

MINISTERUL EDUCAȚIEI, CULTURII ȘI CERCETĂRII
AL REPUBLICII MOLDOVA

CURRICULUM NAȚIONAL

ИНФОРМАТИКА

Clasele X-XII

- Curriculum disciplinar
- Ghid de implementare

Chișinău, 2020

CURRICULUM DISCIPLINAR

Aprobat:

- Consiliul Național pentru Curriculum, proces-verbal nr. 22 din 05.07.2019
- Ordinul Ministerului Educației, Culturii și Cercetării nr. 906 din 17.07.2019

COORDONATORI:

- **Angela CUTASEVICI**, Secretar de Stat în domeniul educației, MECC
- **Valentin CRUDU**, dr., șef Direcție învățământ general, MECC, coordonator al managementului curricular
- **Angela PRISĂCARU**, consultant principal, MECC, coordonator al grupului de lucru

EXPERTI-COORDONATORI:

- **Vladimir GUȚU**, dr. hab., prof. univ., USM, expert-coordonator general
- **Anatol GREMALSCHI**, dr. hab., prof. univ., Institutul de Politici Publice, expert-coordonator pe ariile curriculare *Matematică și științe și Tehnologii*

GRUPUL DE LUCRU:

- **Anatol GREMALSCHI** (coordonator), dr. hab., prof. univ., Institutul de Politici Publice
- **Gheorghe CHISTRUGA**, grad did. superior, IPLT „Mihai Eminescu”, or. Drochia
- **Irina CIOBANU**, grad did. superior, Centrul Tehnologii Informaționale și Comunaționale în Educație
- **Sergiu CORLAT**, grad did. superior, IPLT „Orizont”, Chișinău
- **Svetlana GOLUBEV-BRÎNZĂ**, grad did. superior, specialist principal, metodist, DGETS, mun. Chișinău
- **Lilia MIHĂLACHE**, dr., grad did. superior, IPLT „Ion Creangă”, Chișinău
- **Grigore VASILACHE**, grad did. superior, IPLT „Mircea Eliade”, Chișinău
- **Dorina CHEPTĂNARU**, grad did. întâi, DRÎTS Râșcani, IPLT „Liviu Damian”, or. Râșcani

Traducere: **Irina CIOBANU**, grad did. superior, Centrul Tehnologii Informaționale și Comunaționale în Educație

Информатика : Curriculum național : Clasele 10-12 : Curriculum disciplinar : Ghid de implementare / Ministerul Educației, Culturii și Cercetării al Republicii Moldova ; coordonatori: Angela Cutasevici, Valentin Crudu, Angela Prisăcaru ; grupul de lucru: Anatol Gremalschi (coordonator) [et al.] ; traducere: Irina Ciobanu. – Chișinău : Lyceum, 2020 (F.E.-P. "Tipografia Centrală"). – 120 p. : fig., tab.

Referințe bibliogr.: p. 119-120 (27 tit.). – 300 ex.

ISBN 978-9975-3439-4-7.

373.5.091:004(073)

И 741

GHID DE IMPLEMENTARE

Elaborat în conformitate cu prevederile Curriculumului disciplinar, aprobat la ședința Consiliului Național pentru Curriculum, prin ordinul Ministerului Educației, Culturii și Cercetării nr. 906 din 17.07.2019

COORDONATORI:

- **Angela CUTASEVICI**, Secretar de Stat în domeniul educației, MECC
- **Valentin CRUDU**, dr., șef Direcție învățământ general, MECC, coordonator al managementului curricular
- **Angela PRISĂCARU**, consultant principal, MECC, coordonator al grupului de lucru

EXPERTI-COORDONATORI:

- **Vladimir GUȚU**, dr. hab., prof. univ., USM, expert-coordonator general
- **Anatol GREMALSCHI**, dr. hab., prof. univ., Institutul de Politici Publice, expert-coordonator pe ariile curriculare *Matematică și științe și Tehnologii*

GRUPUL DE LUCRU:

- **Anatol GREMALSCHI** (coordonator), dr. hab., prof. univ., Institutul de Politici Publice
- **Gheorghe CHISTRUGA**, grad did. superior, , IPLT „Mihai Eminescu”, or. Drochia
- **Irina CIOBANU**, grad did. superior, Centrul Tehnologii Informaționale și Comunaționale în Educație
- **Sergiu CORLAT**, grad did. superior, IPLT „Orizont”, Chișinău
- **Svetlana GOLUBEV-BRÎNZA**, grad did. superior, specialist principal, metodist, DGETS, mun. Chișinău
- **Lilia MIHĂLACHE**, dr., grad did. superior, IPLT „Ion Creangă”, Chișinău
- **Grigore VASILACHE**, grad did. superior, IPLT „Mircea Eliade”, Chișinău
- **Dorina CHEPTĂNARU**, grad did. întâi, DRÎTS Râșcani, IPLT „Liviu Damian”, or. Râșcani

Traducere: **Irina CIOBANU**, grad did. superior, Centrul Tehnologii Informaționale și Comunaționale în Educație

Введение

Куррикулум по дисциплине *Информатика*, совместно со школьным учебником, методическим пособием, дидактическим программным обеспечением и др. является составной частью *Национального куррикулума*.

Куррикулум разработан в соответствии с положениями *Кодекса Республики Молдова об образовании (2014)*, *Основ Национального куррикулума (2017)*, *Базового куррикулума: система компетенций для общего образования (2018)* и с *Рекомендациями Европейского парламента и Совета Европейского союза по ключевым компетенциям для обучения в течение всей жизни (Брюссель, 2018 г.)*. Куррикулум по дисциплине *Информатика* является нормативным документом, который единым образом представляет концептуальный, телеологический, содержательный и методологический подходы, ставя акцент на систему компетенций в качестве основы для конечных образовательных целей.

Куррикулум по дисциплине *Информатика* поддерживает и направляет деятельность преподавателей, способствует творческому подходу к долгосрочному и поурочному планированию, а также к самому процессу обучения – учения – оценивания.

Дисциплина *Информатика*, представляемая в данном куррикулуме в дидактическом аспекте, играет важную роль в формировании личности учащихся, в формировании и развитии как цифровых компетенций, так и компетенций для обучения в течение всей жизни, способствует интеграции учащихся в общество, основанное на знаниях.

В процессе разработки Куррикулума по дисциплине *Информатика* учитывались:

- постмодернистский подход и тенденции развития учебных программ на национальном и международном уровне;
- необходимость адаптации дисциплинарной программы к ожиданиям общества, потребностям учащихся и традициям национальной школы;
- возможности дисциплины для формирования трансверсальных, трансдисциплинарных и дисциплинарных компетенций;
- необходимость обеспечения преемственности и взаимосвязи между уровнями и циклами общего образования: дошкольное образование, начальное образование, гимназическое образование и среднее образование.

Основными функциями Куррикулума по дисциплине *Информатика* являются:

- регламентация процесса обучения-учения-оценивания в контексте компетентностной педагогики;
- обеспечение согласованности между информатикой и другими дисциплинами из области *Математика и Естествознание*;
- установление реперов для дидактического проектирования и развития образовательного процесса с точки зрения компетентностной педагогики;

- базовый компонент для разработки стратегии оценивания в информатике;
- ориентация учебного процесса на компетентностную подготовку учащихся;
- фундаментальный компонент для разработки школьных учебников, электронных учебников, методических пособий, оценочных тестов.

Куррикулум содержит следующие компоненты: *Введение; Администрирование дисциплины; Концептуальные основы; Специфические компетенции дисциплины; Учебные единицы*, включающие единицы компетенций, единицы содержания, деятельность и продукты учения; *Методические рекомендации для обучения – учения – оценивания; Библиография.*

Конечные образовательные цели, чётко сформулированные для каждого класса, представляют градуальное развитие специфических компетенций дисциплины и предназначены для определения целей итогового оценивания.

I. Концептуальные основы

Определение школьной дисциплины Информатика. Объектом изучения информатики как межпредметной науки является автоматизированный сбор, накопление, обработка, передача и распространение информации с помощью цифровых устройств.

Как школьная дисциплина информатика участвует в формировании и общем развитии личности учащихся. Основным ее назначением является развитие у учащихся алгоритмического мышления, создание предпосылок для их органической интеграции в современное и будущее информационное общество. Такая интеграция подразумевает, что будущий выпускник гимназии имеет базовые знания в области информатики и обладает навыками использования инструментов, предлагаемых информационно-коммуникационными технологиями, то есть овладение всем набором компетенций, формирующих *информационную культуру* человека.

Статус дисциплины в учебном плане образования. Информатика — это обязательная дисциплина в составе куррикулумной области *Технологии*.

Формативная ценность дисциплины включает:

- знание основных понятий информатики, в том числе элементов логики, алгоритмизации, моделирования и программирования, накопления, хранения и обработки информации;
- формирование практических навыков использования цифровых средств для обработки информации;
- формирование практических навыков общения с использованием цифровых средств передачи и распространения информации.

Принципы, специфичные для преподавания – изучения дисциплины Информатика. Куррикулум по дисциплине *Информатика* для лица предлагает интегрированную модель изучения этой дисциплины. Данная модель способствует формированию у учащихся единой концепции об информатике как науке и о методах реализации информационных концепций для непрерывного развития современного общества. В этом контексте изложены следующие конкретные принципы дисциплины информатики:

Интегрированный подход к дисциплине – структурирование содержания в виде интегрированной модульной модели, целью которой является создание и развитие у учащихся компетенций по использованию информационных систем и непрерывное развитие алгоритмического мышления.

Ориентирование дидактической деятельности на ученика – принятие активной модели обучения, ориентированной на ученика, с использованием индивидуальной или групповой деятельности, которая позволяет развивать самостоятельность, оригинальность, творческий подход, способность работать в команде, учитывая индивидуальность ритма обучения.

Обеспечение социальной функциональности учебного процесса – развитие навыков и компетенций, необходимых для органичной интеграции учащихся в информационное общество. Принцип достигается за счет практической деятельности по изучению и использованию компьютерных программ общего пользования и общения.

Обеспечение междисциплинарных связей – трактовка дидактического подхода с учетом тесной связи со всеми школьными дисциплинами, использование принципов и методов информатики для нахождения решений, разработка проектов, обработка информации, характерной для других школьных предметов, использование цифровых образовательных ресурсов.

Общие рекомендации по обучению – учению дисциплины Информатика. Общий учебно-познавательный процесс информатики разработан в контексте системы компетенций для обучения в гимназии.

Школьная компетенция – это интегрированная система знаний, навыков, отношений и ценностей, приобретенных, сформированных и разработанных в процессе обучения, мобилизация которых позволяет выявлять и решать различные проблемы в различных контекстах и ситуациях.

Система компетенций, входящая в предметный курс информатики, включает:

Ключевые/трансверсальные компетенции, являющиеся важной куррикулярной категорией с высоким уровнем абстрагирования и обобщения, которая отражает ожидания общества в отношении школьного обучения и общие результаты, которые должны быть достигнуты учащимися в конце обучения. Они отражают как тенденции в национальной политике в области образования, изложенные в *Кодексе об образовании* (2014 год), так и тенденции международной политики, изложенные в *Рекомендациях Европейской комиссии* (2018 год).

Ключевые/трансверсальные компетенции относятся к различным сферам социальной жизни и являются мульти-/меж-/трансдисциплинарными.

Специфические компетенции дисциплины вытекают из ключевых/трансверсальных компетенций. Компетенции, специфичные для каждого школьного предмета, представлены в соответствующем Куррикулуме по предмету и, как ожидается, должны сформироваться к концу 12 класса. В школьном предмете *Информатика* они отражены в рамках девяти конкретных компетенций дисциплины, единиц компетенций, модулей содержания, рекомендуемых для обучения видов деятельности и школьных продуктов.

Специфические компетенции дисциплины, разработанные для всех лицейских классов, являются направляющими долгосрочного проектирования процесса обучения. Ежегодный дидактический проект дисциплины разрабатывается в соответствии с данными Администрирования дисциплины и с учетом ориентировочного распределения часов по модулям содержания.

Системы единиц компетенций, спроектированные для единиц обучения, предусмотрены в полном объеме для итогового оценивания в конце этих единиц и выборочно для формативного оценивания. Эти системы являются опорными для дидактического проектирования учебных единиц и для краткосрочного дидактического проектирования.

Системы единиц компетенций, синтезированные в конце каждого класса, предусмотрены для годового итогового оценивания.

Единицы компетенций являются составной частью специфических компетенций и способствуют их формированию, представляя этапы этого процесса.

Единицы компетенций структурированы и разработаны для каждой дисциплины, для каждого из 10-12 классов, для каждого модуля обучения/учебного года и представлены в соответствующем Куррикулуме по предмету.

Модули содержания являются информационным средством для формирования единиц компетенций, предназначенных для соответствующих единиц обучения. Соответственно, имеется в виду формирование компетенций, специфичных для данной дисциплины, а также трансверсальных/трансдисциплинарных компетенций.

Модули содержания включают темы и списки терминов, характерных для предмета: слова/фразы, которые должны входить в активный словарный запас учащегося по завершении этой единицы обучения.

Рекомендуемые для обучения виды деятельности и школьные продукты представляют собой открытый список важных контекстов для демонстрации формирования/развития и оценивания единиц компетенций, предусмотренных в рамках соответствующей единицы обучения. Преподаватель обладает свободой и несёт ответственность за персонализированное использование этого списка для разработки и реализации уроков, также он вправе дополнять его в соответствии с конкретным классом учащихся, имеющимися ресурсами и т. д.

II. Администрирование дисциплины

Статус дисциплины	Куррикулумная область	Класс	Количество модулей по классам	Количество часов в году	
				Гуманитарный профиль	Реальный профиль
Обязательная дисциплина	Технологии	X	5	34	68
		XI	7	34	68
		XII	5	33	66

Дисциплина включает обязательные модули и модули по выбору. В начале каждого учебного года преподаватель поможет учащимся и родителям сориентироваться в определении одного из модулей по выбору, с учетом обеспечения учебного заведения цифровым оборудованием и программными продуктами, необходимыми для изучения данного модуля. Далее на соответствующий учебный год выбранный модуль становится обязательным.

Распределение модулей по классам

Класс	Модули	Примерное число часов		
		Гуманитарный	Реальный	Расширение
X	1. Методы описания естественных и формальных языков	4	8	-
	2. Словарь и синтаксис языка программирования высокого уровня	2	4	-
	3. Концепция данных. Простые типы данных	6	10	-
	4. Концепция действия. Операторы языка программирования высокого уровня	10	24	-
	5. Модуль по выбору: А) Веб-дизайн; Б) Компьютерная графика; В) Цифровая фотография	10	20	-
	<i>На усмотрение преподавателя</i>	2	2	-
	Всего	34	68	-

Класс	Модули	Примерное число часов		
		Гуманитарный	Реальный	Расширение
XI	1. Составные типы данных	10	10	10
	2. Информация	6	10	10
	3. Арифметические основы вычислительной техники	-	6	6
	4. Булева алгебра	-	6	4
	5. Логические схемы	-	6	6
	6. Компьютер и компьютерные сети	6	8	8
	7. Модуль по выбору: А) Техника обработки аудио-визуальной информации; Б) Визуальное программирование; В) Языки гипертекстовой разметки	10	20	20
	<i>На усмотрение преподавателя</i>	2	2	4
	Всего	34	68	68
XII	1. Подпрограммы	6	12	0
	2. Техника программирования	-	12	24
	3. Моделирование и численные методы	-	10	20
	4. Базы данных	16	12	-
	5. Модуль по выбору: А) Углубленные методы обработки информации в базах данных; Б) Экспериментальные методы в гуманитарных науках; В) Веб-программирование; Г) Динамические структуры данных	10	18	18
	<i>На усмотрение преподавателя</i>	1	2	4
	Всего	33	66	66

Последовательность модулей может быть изменена при соблюдении научной и дидактической логики.

III. Специфические компетенции дисциплины

Лицейское образование предусматривает формирование следующих специфических компетенций по *Информатике*:

1. Использование цифровых инструментов для повышения эффективности процесса обучения и работы, демонстрируя инновационный и практический подходы.
2. Взаимодействие с членами виртуальных сообществ в целях обучения и работы, проявляя интерес к активному обучению, исследованиям и сотрудничеству, соблюдая этику виртуальных сред.
3. Распространение в цифровых средах результатов личных и командных достижений, доказывая изобретательность, командный дух и убежденность.
4. Создание цифровых графических, видео и аудио продуктов, демонстрируя креативность и уважение по отношению к национальным и универсальным ценностям.
5. Научное восприятие роли и воздействия информационных явлений на современное общество, проявляя критическое и позитивное мышление при интеграции различных областей познания и человеческой деятельности.
6. Обработка данных экспериментов из области реальных и общественных наук, демонстрируя критическое мышление, ясность и корректность.
7. Алгоритмизация методов анализа, синтеза и нахождения решений проблемных ситуаций, демонстрируя креативность и настойчивость.
8. Внедрение алгоритмов с помощью сред разработки программ, демонстрируя сосредоточенность и настойчивость.
9. Исследование проблемных ситуаций с помощью моделирования, планирования и проведения виртуальных экспериментов в цифровых средах, демонстрируя аналитический подход, ясность и лаконичность.

IV. Единицы обучения

X КЛАСС

Единицы компетенций	Единицы содержания	Рекомендуемая учебная деятельность и школьные продукты
<ul style="list-style-type: none"> - Объяснение способа описания грамматических конструкций через чередование, конкатенации, повторение и возможное включение. - Использование формул БНФ и синтаксических диаграмм для проверки корректности текста. 	<p>1. Методы описания естественных и формальных языков</p> <p>Способы описания грамматических конструкций:</p> <ul style="list-style-type: none"> - метаязык БНФ; - синтаксические диаграммы 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - проверке корректности предложенных текстов с помощью заданных синтаксических диаграмм и металингвистических формул; - написанию текстов с помощью металингвистических формул и синтаксических диаграмм; - написанию металингвистических формул, которые соответствуют заданным синтаксическим диаграммам; - рисованию синтаксических диаграмм, соответствующих предложенным металингвистическим формулам; - применению синтаксических диаграмм и металингвистических формул для проверки корректности предложенных текстов.
<ul style="list-style-type: none"> - Идентификация лексических единиц языка программирования высокого уровня. - Использование формул БНФ и синтаксических диаграмм для проверки корректности текста и лексических единиц. - Применение правил составления идентификаторов, строк символов, чисел, комментариев. 	<p>2. Словарь и синтаксис языка программирования высокого уровня</p> <p>Лексические единицы языка программирования высокого уровня</p> <p>Алфавит языка</p> <p>Словарь языка</p> <p>Лексические единицы:</p> <ul style="list-style-type: none"> - специальные символы; - ключевые слова; - идентификаторы; - разделители; - числа; - строки символов 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - классификации символов языка программирования; - описанию лексических единиц языка программирования; - рисованию синтаксических диаграмм для изучаемых лексических единиц; - нахождению эквивалентных идентификаторов; - представлению переменных в предложенных задачах посредством идентификаторов; - составлению идентификаторов, отображающих специфику задач по физике, математике, химии, обработке текстов и изображениям; - проверке корректности чисел и строк с помощью синтаксических диаграмм и металингвистических формул; - записи вещественных чисел с и без использования масштабного множителя.

3. Концепция данных. Простые типы данных		
<ul style="list-style-type: none"> - Классификация простых типов данных на предопределённые и перечисляемые пользователи порядковые, непорядковые, анонимные и явно определённые типы. - Классификация данных программы на константы и переменные. - Использование идентичных и совместимых типов; - Использование синтаксических диаграмм и металингвистических формул для объявления типов, констант и переменных. 	<p>Концепция данных</p> <p>Определение типов данных:</p> <ul style="list-style-type: none"> - целые; - вещественные; - логические; - символьные; - перечисляемые; - интервальные <p>Идентичные и совместимые типы</p> <p>Порядковые типы данных</p> <p>Анонимные и явно определённые типы данных*</p> <p>Объявление:</p> <ul style="list-style-type: none"> - констант; - типов данных <p>Описание переменных</p>	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - объяснению способа представления данных на языке программирования высокого уровня; - уточнению множества значений и множества допустимых операторов для предложенных типов данных; - определению идентичных (тождественных) и совместимых типов данных; - рисованию синтаксических диаграмм для описания переменных; - определению множества значений, которое может принять переменная, ассоциированная с определённым типом данных; - уточнению операций, которые возможны над значениями переменных заданного типа данных; - рисованию синтаксических диаграмм для определения констант; - определению типов констант в заданных программах.
4. Концепция действия. Операторы языка программирования высокого уровня		
<ul style="list-style-type: none"> - Идентификация составных частей программы. - Использование металингвистических формул и синтаксических диаграмм изучаемого оператора для проверки синтаксической правильности разработанных программ. 	<p>Концепция действия</p> <p>Понятие оператора</p> <p>Вывод алфавитно-цифровой информации на экран</p> <p>Ввод данных с клавиатуры</p> <p>Выражения</p> <p>Вычисление выражений</p> <p>Типы выражений*</p> <p>Операторы:</p>	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - составлению программ, в которых используются стандартные процедуры ввода и вывода; - составлению программ, в которых информация выводится в соответствии с заданными форматами; - классификации операторов языка программирования; - объяснению концепции действия; - рисованию синтаксических диаграмм; - переводу математических выражений на язык программирования высокого уровня и наоборот;

<ul style="list-style-type: none"> - Разработка программ, в которых используются стандартные процедуры чтения/записи. - Разработка программ, в которых используются изучаемые операторы. - Описание известных алгоритмов математических, физических, химических, биологических процессов и явлений, используя различные способы описания алгоритмов. - Разработка алгоритмов для решения задач из различных областей человеческой деятельности. - Перевод разработанных алгоритмов в программы. - Тестирование программ, интерпретация и анализ их результатов. 	<ul style="list-style-type: none"> - вызова процедуры; - пустой оператор; - составной оператор; - оператор присваивания; - условный оператор; - оператор выбора; - оператор цикла со счетчиком; - пока цикла с предусловием; - оператор цикла с постусловием 	<ul style="list-style-type: none"> - использованию стандартных функций языка программирования высокого уровня; - вычислению выражений на языке программирования высокого уровня; - разработке программ для вычисления предложенных выражений и определению их типа; - классификации выражений в зависимости от их типов; - объяснению назначения пустого оператора и разделителей в его представлении; - объяснению способа исполнения операторов: вызова процедуры, присваивания, условного оператора, оператора выбора, повтора со счетчиком, повтора по условию; - интуитивному представлению (через рисунок) структуры разработываемых программ; - определению совместности компонентов операторов с точки зрения присваивания; - разработке программ, в которых используются изучаемые операторы; - описанию известных алгоритмов из курсов математики, физики, химии (например: классификация треугольников в зависимости от сторон/углов, вычисление площадей геометрических фигур, нахождение наибольшего общего делителя и пр.); - разработке алгоритмов для решения задач из различных областей человеческой деятельности; - переводу разработанных алгоритмов в программы на языках программирования высокого уровня; - тестированию программ и анализу их результатов.
---	---	---

5-А. ВЕБ-ДИЗАЙН

<ul style="list-style-type: none"> - Идентификация требований и рекомендаций относительно Веб-документов. - Прохождение этапов разработки Веб-документа. - Разработка Веб-документов с помощью офисных приложений. - Разработка Веб-документов с помощью специализированных приложений. - Разработка Веб-документов с помощью онлайн приложений; - Публикация Веб-документов в среде Интернет. 	<p>Веб-документы:</p> <ul style="list-style-type: none"> - понятия и концепты; - форматы Веб-документов; - структура сайта; - требования к Веб-документам; - этапы разработки Веб-документов; - программные приложения для создания и генерации Веб-документов <p>Объекты из состава документов Веб:</p> <ul style="list-style-type: none"> - текст; - списки; - таблицы; - изображения; - гиперссылки <p>Создание документов Веб при помощи офисных приложений</p> <p>Создание документов Веб при помощи специальных приложений</p> <p>Создание документов Веб при помощи онлайн приложений</p> <p>Публикация документов Веб в Интернете</p>	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - объяснению смысла терминов Веб-документ, сайт; - схематичному представлению через рисунок структуры сайта; - распознаванию этапов разработки Веб-документа и объяснению содержания каждого этапа; - созданию Веб-документов с помощью офисных программ; - созданию Веб-документов с помощью специализированных программ; - созданию Веб-документов с помощью онлайн приложений; - публикации Веб-документов в среде Интернет. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Анализ часто посещаемых сайтов с точки зрения дизайна, функциональности и удобства использования. <p><i>Проекты (примеры):</i></p> <ul style="list-style-type: none"> - Формулирование требований к сайту ("Моя школа", "Мой город", "Родное село", "Магазин"); - Разработка структуры сайта ("Моя школа", "Мой город", "Родное село", "Магазин"); - Создание сайта ("Моя школа", "Мой город", "Родное село", "Магазин").
--	--	--

<ul style="list-style-type: none"> - Соблюдение национального и международного законодательства в области авторских прав. 	<p>Элементы права:</p> <ul style="list-style-type: none"> - авторские права; - смежные права; - виды лицензий (Creative Commons; - антиплагиат в области мультимедиа 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - идентификации признаков декларирования авторских прав; - объяснению правил соблюдения авторских прав; - использованию лицензий Creative Commons.
5-В. Компьютерная графика		
<ul style="list-style-type: none"> - Идентификация элементов растровых изображений. - Создание и редактирование растровых изображений. - Импорт и экспорт графических файлов. - Использование техник и сервисов Веб для редактирования, хранения и распространения цифровых изображений. 	<p>Растровые изображения:</p> <ul style="list-style-type: none"> - создание растровых изображений; - импорт растровых изображений; - объекты; - слои; - спецэффекты <p>Редактирование растровых изображений:</p> <ul style="list-style-type: none"> - клонирование; - упорядочение; - группирование; - масштабирование и вращение; - размытие; - применение фильтров; - применение тектуры <p>Хранение растровых изображений:</p> <ul style="list-style-type: none"> - локальные альбомы и архивы; - Веб альбомы и архивы 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - идентификации элементов растровых изображений; - созданию и редактированию растровых изображений; - импорту и экспорту растровых изображений; - модификации графического формата растровых изображений; - визуализации серий изображений; - созданию собственного графического профиля на специализированных платформах Веб; - созданию локальных и Веб альбомов изображений; - созданию локальных/Веб архивов; - управлению локальными/Веб архивами. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Эволюция бесплатных растровых графических редакторов; - Эволюция коммерческих растровых графических редакторов; - Flickr – платформа Веб, ориентированная на изображения; - Picasa – инструмент Google для организации в Веб цифровых альбомов. <p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Конкурсы тематических постеров; - Создание цифровой коллекции дорожных знаков; - Разработка тематических цифровых фотоальбомов: "Моя школа", "Моё село", "Мои друзья".

<ul style="list-style-type: none"> - Идентификация элементов векторного изображения. - Создание векторных изображений. - Векторизация растровых изображений. 	<p>Векторные изображения:</p> <ul style="list-style-type: none"> - графические примитивы; - сетка; - операции над графическими примитивами. <p>Редактирование векторных изображений:</p> <ul style="list-style-type: none"> - инструменты редактирования; - настройка контура; - настройка тени; - настройка 3D; - настройка прозрачности; - клонирование; - упорядочивание <p>Импорт и экспорт векторных изображений</p>	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - идентификации элементов и свойств векторных изображений; - созданию и редактированию векторных изображений; - импорту векторных изображений; - экспорту векторных изображений. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Эволюция бесплатных векторных графических редакторов; - Эволюция коммерческих векторных графических редакторов.
<ul style="list-style-type: none"> - Соблюдение национального и международного законодательства в области авторских прав. 	<p>Элементы права:</p> <ul style="list-style-type: none"> - авторские права; - смежные права; - виды лицензий Creative Commons; - антиплагат в области мультимедиа 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - идентификации признаков декларирования авторских прав; - объяснению правил соблюдения авторских прав; - использованию лицензий Creative Commons.

5-С. Цифровая фотография		
<ul style="list-style-type: none"> - Уточнение ключевых понятий цифровой фотографии. - Идентификация главных компонентов цифровой фотокамеры. - Выбор подходящего типа цифровой фотокамеры для планируемых действий. 	<p>Базовые понятия цифровой фотографии:</p> <ul style="list-style-type: none"> - размеры; - разрешение; - разрешающая способность; - цветовая модель; - графические форматы <p>Компоненты цифровых фотокамер</p> <p>Категории цифровых фотокамер</p> <p>Использование цифровых фотокамер в различных условиях и средах</p> <p>Оборудование для фотографирования:</p> <ul style="list-style-type: none"> - объективы; - вспышки; - штативы; - инструменты и приспособления 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - идентификации свойств цифровых фотографий; - идентификации компонентов цифровых фотокамер; - фотографированию и последующему управлению файлами с помощью фотокамеры. <p><i>Задачи по:</i></p> <ul style="list-style-type: none"> - вычислению количества информации в цифровых фотографиях; - вычислению возможного разрешения цифровых фотографий; - определению корреляции между размерами экрана и размерами отпечатанной фотографии. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - История фотографии; - История цифровой фотографии; - Особенности зеркальных цифровых фотокамер DSLR; - Факторы, влияющие на качество цифровой фотографии.
<ul style="list-style-type: none"> - Знание главных факторов, которые влияют на качество процесса фотосъемки. - Применение стандартного режима для различных типов фотографий. 	<p>Факторы управления:</p> <ul style="list-style-type: none"> - композиция; - фокус; - глубина резкости; - экспозиция (выдержка) 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - фотографированию и кадрированию; - формированию кадра по "золотому правилу"; - центрированию объектов в кадре; - уравниванию элементов фотографии; - фокусировке при фотографировании портретов, пейзажей, близких предметов;

<ul style="list-style-type: none"> - Выбор оптимальных параметров для персонализированного режима съемки. - Фотографирование в особом режиме/условиях. 	<p>Техники цифровой фотографии для различных сюжетов:</p> <ul style="list-style-type: none"> - портрет и люди; - натюрморт; - репортаж; - архитектура; - пейзаж; - спорт; - животные; - абстрактные фотографии 	<ul style="list-style-type: none"> - фотографированию с размытием фона; - правильному выбору экспозиции, ISO; - установке баланса белого; - выбору переднего плана; - фотографированию портретов; - фотографированию натюрмортов; - фотографированию репортажей; - фотографированию в путешествии; - фотографированию архитектурных объектов; - фотографированию пейзажей; - фотографированию спортивных соревнований; - фотографированию динамических сцен; - фотографированию животных; - фотографированию абстракций. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Топ 10 самых удачных фотопортретов; - Животные в природе; - Самые экзотические цветы; - Фоторепортажи журналистов; - Карнавал в Рио в фотографиях. <p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Красная Книга в иллюстрациях; - Открой заново свое село в фотографиях; - Как я провел каникулы. Фоторепортаж; - "У нас играют в футбол!"
--	--	---

<p>- Идентификация и использование специализированного инструментария для редактирования цифровых фотографий.</p> <p>- Использование информационного инструментария для хранения и распространения цифровых фотографий.</p>	<p>Техники обработки цифровых фотографий</p> <p>Преобразования:</p> <ul style="list-style-type: none"> - формата; - геометрические; - цветные; - художественные <p>Программные продукты для обработки цифровых фотографий</p> <p>Хранение и распространение цифровых фотографий:</p> <ul style="list-style-type: none"> - локально; - в Интернете 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - импорту фотографий в компьютер; - редактированию фотографий в графическом редакторе; - преобразованию графического формата фотографий; - изменению размеров фотографий; - вырезанию фрагментов из фотографий; - монтажу фрагментов из различных фотографий; - изменению цветовой модели фотографии; - изменению контраста фотографии (ее фрагмента); - изменению кривых интенсивности первичных цветов; - исключению эффекта красных глаз; - добавлению динамического элемента к фрагменту изображения; - созданию панорамных фотографий; - изменению источников света в фотографии; - применению эффекта витража; - применению эффекта кристаллизации; - применению эффекта ветра; - применению эффекта дождя; - применению эффекта постера. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Самые популярные художественные эффекты в фотографии; - Топ приложений для редактирования цифровых фотографий; - Flickr – платформа Web для изображений; - Picasa – инструмент google для организации в цифровых Веб-альбомах; <p><i>Проекты:</i></p> <p>Разработка цифровых тематических альбомов:</p> <ul style="list-style-type: none"> - "Моя школа"; - "Моё село"; - "Мои друзья".
---	---	--

<ul style="list-style-type: none"> - Соблюдение национального и международного законодательства в области авторских прав. 	<p>Элементы права:</p> <ul style="list-style-type: none"> - авторские права; - смежные права; - виды лицензий Creative Commons; - антиплагиат в области мультимедиа 	<p>Упражнения по:</p> <ul style="list-style-type: none"> - идентификации признаков декларирования авторских прав; - объяснению правил соблюдения авторских прав; - использованию лицензий Creative Commons.
--	---	--

* Только для реального профиля.

По окончании X класса ученик может:

- проверять правильность лексических единиц и формальных текстов с помощью синтаксических диаграмм и металингвистических формул;
- создавать синтаксические диаграммы на основе металингвистических формул и выводить металингвистические формулы на основе синтаксических диаграмм;
- выбирать типы данных в зависимости от специфики информации из решаемой задачи;
- разрабатывать и отлаживать на компьютере алгоритмы обработки неструктурированных (простых) данных;
- интерпретировать результаты выполнения компьютерных программ, *демонстрируя следующие преобладающие специфические отношения:*
 - правильность и когерентность информационного языка;
 - внимание и интерес к реализации алгоритмов;
 - инициативу и настойчивость при реализации алгоритмов;
 - использование критического мышления при принятии плана алгоритмизации проблем;
 - любопытство и креативность при интеграции знаний и навыков из области информатики со знаниями и навыками из других областей;
- соблюдение правил безопасности, эргономики, этики и дизайна при создании и распространении компьютерных программ.

Единицы компетенций	Единицы содержания	Рекомендуемая учебная деятельность и школьные продукты
<ul style="list-style-type: none"> - Аргументация необходимости структурирования данных. - Распознавание и использование металингвистических формул и синтаксических диаграмм объявлений изучаемых типов данных. - Обработка структурированных данных. - Выбор структур данных, адекватных для решения поставленной задачи. 	<p style="text-align: center;">1. Составные типы данных</p> <p>Составные типы данных (массив, строка символов, запись, файл):</p> <ul style="list-style-type: none"> - объявления типов структурированных данных; - металингвистические формулы и синтаксические диаграммы объявлений структурированных типов данных; - множество значений структурированных типов данных; - ограничения, обусловленные реализациями языков программирования в случае использования структурированных типов данных; - операции, предназначенные для обработки структурированных типов данных; - классификация файлов по типу разрешённых операций и по типу доступа 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - использованию металингвистических формул и синтаксических диаграмм для проверки правильности объявления структурированных типов данных; - объявлению структурированных типов данных; - обработке структурированных типов данных; - обработке файлов. <p><i>Задачи по:</i></p> <ul style="list-style-type: none"> - обработке данных, используя изучаемые типы данных. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Метод ссылки на компоненты структурированных данных; - Методы сохранения структурированных данных на внутренних и внешних носителях информации в компьютере. <p><i>Проекты (примеры):</i></p> <ul style="list-style-type: none"> - Учёт ежедневного потребления электричества в домашних условиях; - Учёт посещаемости в классе; - Вычисление средних оценок учащихся класса; - Вычисление персональных расходов: ежедневных, недельных и по месяцам; - Обработка текстов.

2. Информация		
<ul style="list-style-type: none"> - Идентификация базовых концептов представления и передачи информации. - Использование инструментов информатики в повседневной деятельности. - Оценка количества информации, переданной, принятой и обработанной в повседневной деятельности. - Применение основных принципов кодирования и декодирования информации для ее передачи, получения и обработки. 	<p>Информация:</p> <ul style="list-style-type: none"> - количество информации; - накопление информации; - передача информации; - обработка информации <p>Символы и алфавиты</p> <p>Кодирование и декодирование сообщений</p> <p>Информация непрерывных сообщений*:</p> <ul style="list-style-type: none"> - Дискретные сообщения; - непрерывные сообщения; - аналого-числовой преобразователь; - цифро-аналоговый преобразователь <p>Дискретизация изображений</p> <p>Представление и передача информации</p>	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - определению количества информации в сообщениях источника; - кодированию и декодированию информации; - определению количества информации в текстах, изображениях, аудио- и видеофрагментах; - идентификации источников, каналов и носителей информации; - определению емкости носителей информации; - дискретизации непрерывных сообщений; - дискретизации в пространстве и по значению статистических изображений; - дискретизации по времени динамических изображений. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Эволюция статических и динамических носителей информации; - Области использования современных носителей информации. <p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Оценка количества информации; - Архивирование информации; - Особенности кодирования специальных символов центрально-европейских алфавитов; - Троичное кодирование информации.
3. Арифметические основы вычислительной техники*		
<ul style="list-style-type: none"> - Идентификация пределов представления чисел в компьютере. - Применение компьютерной арифметики для решения задач обработки информации. 	<p>Системы счисления</p> <p>Позиционные и непозиционные системы счисления</p> <p>Позиционные системы счисления:</p> <ul style="list-style-type: none"> - двоичная система; - восьмеричная система; - шестнадцатеричная система 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - аргументации необходимости применения в информатике компьютерной арифметики; - идентификации типа системы счисления; - конверсии чисел из одной системы счисления в другую; - представлению чисел в прямом, обратном и комплементарном коде; - представлению действительных чисел в формате с фиксированной и плавающей точкой;

<p>- Применение основных операций компьютерной арифметики для решения задач из области информатики.</p>	<p>Конверсия чисел из одной системы счисления в другую:</p> <ul style="list-style-type: none"> - из системы по основанию 2, 8, 16 в десятичную систему счисления и наоборот; - из двоичной в восьмеричную, шестнадцатеричную и наоборот <p>Арифметические операции в двоичной системе счисления**:</p> <ul style="list-style-type: none"> - сложение двоичных чисел; - вычитание двоичных чисел <p>Представление чисел в компьютере**:</p> <ul style="list-style-type: none"> - целые числа в прямом, обратном и дополнителем (комплементарном) коде; - вещественные числа с фиксированной и плавающей точкой 	<ul style="list-style-type: none"> - применению терминологии, соответствующей компьютерной арифметике; - аргументации использования в информатике специализированной компьютерной арифметики. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Достоинства и недостатки позиционных и непозиционных систем счисления; - Области использования позиционных и непозиционных систем счисления; - Вычислительная арифметика, внедренная в цифровые устройства повседневногo пользования; - Достоинства и недостатки представления чисел в формате с фиксированной и плавающей точкой. <p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Внедрение компьютерной арифметики в цифровые аппараты для воспроизведения звука; - Внедрение компьютерной арифметики в цифровые аппараты для воспроизведения видео; - Троичная компьютерная арифметика.
<p>4. Булева алгебра*</p>		
<ul style="list-style-type: none"> - Применение булевой алгебры для цифровой обработки информации. - Аргументация необходимости использования булевой алгебры в информатике. 	<p>Логические переменные и выражения:</p> <ul style="list-style-type: none"> - логические константы и переменные; - логические операции НЕТ, И, ИЛИ; - логические выражения; 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - применению специальной терминологии из области булевой алгебры; - объяснению роли булевой алгебры в описании и проектировании цифровых устройств; - созданию таблиц истинности логических выражений; - заданию логических функций посредством таблиц и формул; - составлению таблиц истинности логических функций.

	<ul style="list-style-type: none"> - таблицы истинности логических выражений <p>Логические функции:</p> <ul style="list-style-type: none"> - методы задания логических функций; - наиболее часто используемые логические функции; - полная система логических функций 	<p><i>Задачи по:</i></p> <ul style="list-style-type: none"> - составлению таблиц истинности логических выражений с/без написания программ на языке программирования высокого уровня; - вычислению значений логических функций.
5. Логические схемы*		
<ul style="list-style-type: none"> - Представление логических переменных посредством физических величин. - Применение иерархического проектирования для разработки вычислительных структур, обработки и хранения данных. - Материализация логических функций. 	<p>Элементарные логические схемы</p> <p>Классификация логических схем</p> <p>Часто используемые логические вентили</p> <p>Часто используемые комбинационные схемы:</p> <ul style="list-style-type: none"> - сумматор; - компаратор; - шифратор; - дешифратор; - мультиплексор; - демультиплексор <p>Часто используемые последовательные схемы:</p> <ul style="list-style-type: none"> - триггер; - регистр; - счетчик; - генератор импульсов 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - описанию методов представления двоичных значений посредством физических величин; - объяснению влияния способа представления логических переменных на производительность вычислительных систем; - объяснению методов реализации логических функций НЕТ, И, ИЛИ посредством схем с коммутационными элементами; - выявлению и объяснению связи между формулами логических функций и их реализацией посредством соединения логических вентилей; - объяснению метода функционирования часто используемых комбинационных схем; - объяснению метода функционирования часто используемых последовательных схем; - наблюдению и пониманию связи между представлением данных в компьютере и использованием логических схем; - разработке логических схем на базе логических функций, определенных с помощью таблиц; - разработке логических схем на базе логических функций, определенных с помощью формул.

		<p><i>Тематическое исследование:</i></p> <ul style="list-style-type: none"> - Представление двоичных значений посредством различных значений физических величин. <p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Разработка комбинационных схем, реализующих предложенные логические функции; - Обработка данных с помощью часто используемых логических схем; - Разработка программ по моделированию комбинационных схем.
6. Компьютер и компьютерные сети		
<ul style="list-style-type: none"> - Использование программных и аппаратных ресурсов компьютера. - Использование внешней носителей памяти для текущего и долгосрочного хранения данных. - Использование компьютерных устройств ввода/вывода. 	<p>Функциональная схема компьютера:</p> <ul style="list-style-type: none"> - процессор; - внутренняя память; - устройства ввода/вывода; - внешняя память <p>Принцип программного управления*:</p> <ul style="list-style-type: none"> - данные и команды; - формат команд; - выполнение команд <p>Ресурсы компьютера:</p> <ul style="list-style-type: none"> - устройства; - программы <p>Внешняя память:</p> <ul style="list-style-type: none"> - на магнитных носителях; - на оптических носителях; - на полупроводниках 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - идентификации функциональных устройств компьютера и потоков данных между ними; - объяснению принципа программного управления; - описанию функциональной схемы компьютера; - описанию роли внедрения эффективных алгоритмов в развитии информатики; - классификации команд в зависимости от их типа; - распознаванию программных и аппаратных ресурсов компьютера; - объяснению принципов действия внешней памяти; - объяснению принципов действия устройств ввода/вывода; - классификации компьютеров в зависимости от технико-экономических показателей и областей применения. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Хранение данных на магнитных и оптических носителях; - Хранение данных на оптических и полупроводниковых носителях; - Универсальные и специализированные компьютеры.

<ul style="list-style-type: none"> - Классификация компьютерных сетей. - Использование кооперативных технологий в сети. - Описание принципов организации и использования сервисов Интернета; - Соблюдение правил этики и личной безопасности в компьютерных сетях. - Соблюдение правил информационной защиты сетей. 	<p>Устройства ввода/вывода:</p> <ul style="list-style-type: none"> - монитор; - клавиатура; - мышь; - принтеры <p>Классификация компьютеров</p>	<p><i>Проекты:</i></p> <ul style="list-style-type: none"> - История вычислительной техники; - Эволюция вычислительной техники; - Микропроцессоры; - Компьютеры вокруг нас; - Как выбрать персональный компьютер; - Эволюция структуры компьютеров.
<ul style="list-style-type: none"> - Классификация компьютерных сетей. - Использование кооперативных технологий в сети. - Описание принципов организации и использования сервисов Интернета; - Соблюдение правил этики и личной безопасности в компьютерных сетях. - Соблюдение правил информационной защиты сетей. 	<p>Типы сетей:</p> <ul style="list-style-type: none"> - локальные сети; - региональные сети; - глобальные сети <p>Кооперативные технологии в сети:</p> <ul style="list-style-type: none"> - от равного к равному; - клиент – сервер <p>Топология и архитектура сетей*:</p> <ul style="list-style-type: none"> - пакет данных; - протокол коммуникации <p>Сеть Интернет. Сервисы Интернета:</p> <ul style="list-style-type: none"> - e-коммуникации (e-мейл, форум, чат, социальные сети); - доступ к удаленным компьютерам; - передача файлов; - локализация и поиск информации 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - объяснению принципов функционирования компьютерных сетей; - описанию концепта компьютерной сети; - классификации сетей в зависимости от топологии соединения компьютеров; - классификации сетей в зависимости от расстояния между компьютерами; - наблюдению и пониманию связи между техническими компонентами сети и предоставляемыми ею сервисами; - аргументации необходимости стандартизации протоколов; - объяснению понятия "архитектура компьютерной сети"; - объяснению принципов организации сервисов Интернета. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Возможности сред передачи данных: кабель, оптоволоконно, радиоканал; - Структура компьютерной сети школьного кабинета информатики; - Тип подсоединения лица к сети Интернет и пропускная способность этого соединения; - Тип домашнего подсоединения к сети Интернет и пропускная способность этого соединения. <p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Эволюция компьютерных сетей; - Эволюция сети Интернет; - Сервисы Интернета, доступные в лицейской сети; - Сервисы Интернета, доступные в домашних условиях.

7-А. Техники обработки аудиовизуальной информации		
<ul style="list-style-type: none"> - Использование методов оцифровки аудиоинформации. - Использование аудиоинверторов. - Использование основных форматов аудиофайлов. 	<p>Оцифровка аудиоинформации:</p> <ul style="list-style-type: none"> - звук как физический феномен; - параметры, характеризующие восприятие человеком звуковых волн; - цифровое кодирование аудиоинформации; - компрессия и декомпрессия аудиоинформации <p>Воспроизведение аудиоинформации. Поток данных с диска (битрейт)</p> <p>Программы и оборудование для кодирования и декодирования аудиоинформации</p>	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - объяснению принципов кодирования и декодирования аудиоинформации; - определению объема несжатых аудиоданных на основе параметров кодирования; - определению формата аудиофайла по расширению его имени; - идентификации и объяснению базовых параметров и основных характеристик распространённого оборудования для сбора, записи, обработки и воспроизведения звука. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Сравнительный анализ программ для цифровой обработки звука; - Сравнительный анализ оборудования для цифровой обработки звука; - Сравнительный анализ форматов аудиофайлов. <p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Исследование влияния частоты дискретизации аудио на качество воспроизведения; - Исследование влияния уровня глубины квантования звука на качество воспроизведения; - Исследование влияния уровня сжатия звуковых сигналов на качество воспроизведения.
<ul style="list-style-type: none"> - Использование базовых возможностей цифровых звуковых редакторов. - Использование звуковых редакторов для обработки звука. - Создание и обработка аудиозаписей. 	<p>Цифровая обработка звука</p> <p>Запись и воспроизведение звука</p> <p>Визуализация звуковых сигналов</p> <p>Обработка звукового сигнала:</p> <ul style="list-style-type: none"> - нарезка и компоновка звуковых фрагментов; 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - использованию основных возможностей некоторого цифрового звукового редактора; - сравнительному анализу одних и тех же аудиозаписей, сделанных в разных звуковых форматах; - анализу качества воспроизведения аудиозаписей в зависимости от параметров записи и хранения; - преобразованию форматов аудиофайлов;

	<ul style="list-style-type: none"> - преобразование амплитуды (усиление, смешивание, нормализация); - изменение тональности и длительности воспроизведения; - применение эффектов <p>Анализ звука:</p> <ul style="list-style-type: none"> - спектральный анализ; - анализ амплитуды <p>Преобразование форматов звуковых записей</p>	<ul style="list-style-type: none"> - передаче аудиофайлов при помощи различных средств цифровой коммуникации. <p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Разработка звукового сопровождения к какому-либо школьному мероприятию; - Смешивание звуковых сигналов от различных источников во время школьного мероприятия; - Разработка аудиотреков для видеофильмов школьных мероприятий.
<ul style="list-style-type: none"> - Использование методов оцифровки видеоинформации. - Знание принципов функционирования и применения видеоконверторов. - Знание и применение основных форматов видеофайлов. 	<p>Оцифровка видеоинформации:</p> <ul style="list-style-type: none"> - свет как физический феномен; - параметры, характеризующие восприятие человеком электромагнитных волн; - частота дискретизации во времени, - частота дискретизации в пространстве, - шаг квантования видеосигнала; - разрешающая способность; - число базовых цветов <p>Компрессия и декомпрессия видеоданных</p>	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - описанию областей человеческой деятельности, в которой используется видеоинформация; - описанию и объяснению параметров электромагнитных волн с точки зрения восприятия их человеком; - описанию и объяснению факторов видео, которые представляют опасность для здоровья человека; - объяснению принципов кодирования и декодирования видеоданных; - определению объема несжатых видеоданных на основе известных параметров кодирования; - определению формата видеофайла по расширению его имени; - идентификации и объяснению основных параметров и главных характеристик распространённого оборудования для сбора, записи, обработки и воспроизведения видеозаписей. <p><i>Тематическое исследование:</i></p> <ul style="list-style-type: none"> - Восприятие различных электромагнитных волн человеком; - Влияние изменения параметров электромагнитных волн на их восприятии человеком;

<ul style="list-style-type: none"> - Использование основных возможностей цифровых видеоредакторов. - Использование цифровых видеоредакторов для обработки видеозаписей. - Создание и обработка видеозаписей. 	<p>Воспроизведение видеоинформации. Поток данных с диска (битрейт)</p> <p>Программы и оборудование для кодирования и декодирования видеоданных</p> <p>Форматы видеофайлов</p>	<ul style="list-style-type: none"> - Сравнительный анализ программ для обработки цифровой видеоинформации; - Сравнительный анализ оборудования для обработки цифровой видеоинформации; - Сравнительный анализ форматов видеофайлов. <p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Исследование влияния вариации частоты дискретизации видеосигнала на качество воспроизведения видео; - Исследование влияния шага квантования видеосигнала на качество воспроизведения видео; - Исследование влияния компрессии (сжатия) видеосигнала на качество воспроизведения видео.
<ul style="list-style-type: none"> - Использование основных возможностей цифровых видеоредакторов. - Использование цифровых видеоредакторов для обработки видеозаписей. - Создание и обработка видеозаписей. 	<p>Обработка видеоинформации</p> <p>Запись и воспроизведение видео</p> <p>Обработка видеозаписей:</p> <ul style="list-style-type: none"> - нарезка и склейка видеофрагментов; - монтаж видеофрагментов; - преобразование во времени; - связь с аудиотреком; - изменение длительности воспроизведения; - фильтрация видеосигналов; - применение эффектов; - субтитрование видеозаписей; - преобразование форматов 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - использованию главных возможностей цифрового видеоредактора; - сравнительному анализу видеозаписи, выполненной в различных форматах; - анализу качества воспроизведения видеозаписи в зависимости от параметров записи и хранения; - преобразованию форматов видеофайлов; - передаче видеофайлов с использованием различных средств цифровой коммуникации; - поиску информации мультимедиа в Интернете; - регистрации и созданию собственного профиля в онлайн сервисе; - публикации собственных разработок мультимедиа; - созданию на Веб-страницах гиперссылок с собственными мультимедиа ресурсами, доступными онлайн. <p><i>Тематическое исследование:</i></p> <ul style="list-style-type: none"> - Сравнительный анализ достоинств и недостатков онлайн сервисов для информации мультимедиа.

	<p>Онлайн службы распространения мультимедийной информации:</p> <ul style="list-style-type: none"> - каталог сервисов; - возможности, предлагаемые популярными сервисами; - поиск нужных мультимедийных ресурсов; - распространение собственных мультимедийных продуктов; - редактирование онлайн мультимедийных файлов. 	<p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Разработка видео сопровождения для какого-либо школьного мероприятия; - Смешивание видеосигналов от различных источников в рамках школьного мероприятия; - Монтаж видео, снятого на школьном мероприятии.
<ul style="list-style-type: none"> - Соблюдение национального и международного законодательства в области авторских прав. 	<p>Элементы права:</p> <ul style="list-style-type: none"> - авторские права; - смежные права; - виды лицензий (CC - Creative Commons License); - антиплагат в области мультимедиа 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - идентификации признаков декларирования авторских прав; - объяснению правил соблюдения авторских прав; - использованию различных видов лицензий Creative Commons.
7-В. Визуальное программирование		
<ul style="list-style-type: none"> - Применение базовых понятий визуального программирования. - Уточнение принципов визуального программирования. 	<p>Базовые понятия визуального программирования:</p> <ul style="list-style-type: none"> - графические объекты; - связи; - действия <p>Типы графических объектов:</p> <ul style="list-style-type: none"> - первичные объекты; - метаобъекты 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - объяснению принципов визуального программирования; - объяснению базовых понятий визуального программирования; - классификации объектов/графических инструментов; - контролю событий, действий, состояний. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Графические команды в сравнении с письменными командами; - Гибкость визуального программирования; - Отладка программ: визуальная и процедурная; - Отладка программ: визуальная и объектно-ориентированная.

	<p>Элементы объектно-ориентированного программирования в визуальном контексте:</p> <ul style="list-style-type: none"> - объекты; - свойства <p>Элементы событийно-ориентированного программирования в визуальном контексте:</p> <ul style="list-style-type: none"> - ситуации; - действия; - результаты 	
<ul style="list-style-type: none"> - Идентификация сред визуального программирования. - Использование элементов контроля интерфейса визуальных сред. - Группировка визуальных команд в программы. 	<p>Среды визуального программирования</p> <p>Главные компоненты среды визуального программирования</p> <p>Структура проекта</p> <p>Файлы из состава некоторого проекта</p> <p>Ввод и редактирование программ</p> <p>Отладка и запуск программ</p>	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - классификации сред визуального программирования; - идентификации главных компонентов среды визуального программирования; - объяснению структуры проекта в рамках сред визуального программирования; - идентификации файлов проекта; - запуску программ, разработанных в среде визуального программирования. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Достоинства и недостатки сред визуального программирования по сравнению со средами процедурного программирования; - Сравнительный анализ визуальных и традиционных сред программирования; - Презентация некоторых широко распространенных сред и языков визуального программирования (Alice, Kodu, Scratch, ToonTalk, Cameleon, Filter Forge, пр.).

<ul style="list-style-type: none"> - Идентификация графических команд. - Классификация команд. - Адекватное применение графических команд для создания визуальных программных продуктов. - Создание собственных графических команд для расширения/гибкости возможностей программирования. 	<p>Блоки для визуального программирования</p> <p>Движение:</p> <ul style="list-style-type: none"> - идти; - повернуть; - проверка появления/некоторого события/ситуации <p>Внешний вид:</p> <ul style="list-style-type: none"> - изменить размер; - показать сообщение/значение; - изменить объект; - настроить видимость; - настроить старт <p>Управление звуком:</p> <ul style="list-style-type: none"> - настроить звук; - старт звука; - остановить звук; - длительность звука; - применить звуковые эффекты <p>События:</p> <ul style="list-style-type: none"> - старт/стоп программы; - нажатие клавиш; - нажатие кнопки мыши; - нажатие графического объекта; - контроль потока сообщений 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - перемещению и управлению графическими объектами; - выводу сообщений; - настройке свойств графических объектов: геометрическая форма/цвет/видимость; - интегрированию звуковых элементов (мультимедиа) в команды визуальных программ; - контролю динамических графических объектов с помощью клавиатуры и мыши; - созданию и применению простых подпрограмм. <p><i>Визуальные проекты:</i></p> <ul style="list-style-type: none"> - Взаимодействие двух или более визуальных персонажей (объектов); - Моделирование диалога между персонажами; - Простые игры (теннис, ловля падающих предметов, отслеживание); - Иллюстрация сказок.
---	--	--

<ul style="list-style-type: none"> - Уточнение опций доступа ко внешним данным. - Определение простых и структурированных данных. - Управление данными: операции, ввод, вывод. 	<p>Команды контроля:</p> <ul style="list-style-type: none"> - команды выбора; - команды повторения; - пауза; - стоп; - клонировать <p>Контроль ситуаций:</p> <ul style="list-style-type: none"> - свойства объекта; - ввод данных; - свойства периферических устройств; - таймер <p>Подпрограммы:</p> <ul style="list-style-type: none"> - создание; - вызов 	
	<p>Обработка внешних данных</p> <p>Определение:</p> <ul style="list-style-type: none"> - простые переменные; - структуры данных <p>Операции:</p> <ul style="list-style-type: none"> - арифметические; - реляционные (отношения); - логические <p>Визуальные объекты для:</p> <ul style="list-style-type: none"> - ввода данных; - вывода данных <p>События и действия для изменения данных:</p> <ul style="list-style-type: none"> - программные ситуации; - контроль клавиатуры; - действия мышью 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - определению и контролю простых переменных; - интегрированию простых, логических операций, операций отношения в визуальные программы; - изменению значений переменных путем действий и событий (нажатием клавиш клавиатуры и мыши и др.); - выводу результатов путем их ассоциации с некоторым графическим объектом; - определению и контролю структур данных (array); - поиску в структурах данных; - изменению структур данных. <p><i>Визуальные проекты:</i></p> <ul style="list-style-type: none"> - Наибольшее число из названных персонажами (объектами); - Сумма чисел, названных персонажами; - Проверка простых чисел; - Наибольший общий делитель; - Максимальный по значению элемент массива. <p><i>Тематическое исследование:</i></p> <ul style="list-style-type: none"> - Визуальное программирование вычислительных проектов: преимущества и недостатки.

7-С. Языки гипертекстовой разметки

<ul style="list-style-type: none"> - Распознавание элементов языка HTML. - Создание простых веб-документов. - Форматирование текста инструментами HTML. 	<p>Язык гипертекстовой разметки (HTML)</p> <p>Элементы контроля гипертекста</p> <p>Общая структура HTML документа</p> <p>Форматирование текста:</p> <ul style="list-style-type: none"> - заголовки; - параграфы; - комментарии; - физические стили; - логические стили; - разделители; - списки с нумерацией/без нумерации 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - созданию документа Веб-инструментами HTML; - форматированию текста инструментами HTML; - организации списков инструментами HTML. <p><i>Проекты (этап разработки первичных компонентов):</i></p> <ul style="list-style-type: none"> - Создание документов Веб, используя инструменты HTML – составные части некоторого сайта ("История моего города/села", "Птицы Центрального парка", "Статистика погоды" и др.).
<ul style="list-style-type: none"> - Создание внутренних и внешних ссылок в документах Веб. - Организация навигации между документами Веб. - Организация просмотра общедоступных веб-сайтов. 	<p>Ссылки:</p> <ul style="list-style-type: none"> - внешние, внутренние; - путь; - якорь; - комментарий к ссылке 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - созданию ссылок на внешний документ (находящийся в том же либо другом каталоге/папке); - созданию ссылок на некоторый сайт; - созданию ссылок на некоторый фрагмент того же либо другого документа; - созданию возможности запустить на выполнение некоторого приложения для отправки сообщений; - созданию ссылок на некоторый файл произвольного формата с целью создания на диске копии этого файла. <p><i>Проекты (этап интеграции):</i></p> <ul style="list-style-type: none"> - Создание ссылок на документы HTML – составные части некоторого сайта ("История моего города/села", "Птицы Центрального парка", "Статистика погоды" и др.).

<ul style="list-style-type: none"> - Вставка ссылок на мультимедийные объекты в веб-документах. - Интеграция мультимедийного контента в веб-документы. 	<p>Внешние мультимедийные объекты</p> <p>Изображения/видео:</p> <ul style="list-style-type: none"> - размеры; - контур; - ссылки; - комментарии; - контроль <p>Звук:</p> <ul style="list-style-type: none"> - ссылки; - комментарии; - контроль <p>Встроенные объекты мультимедиа:</p> <ul style="list-style-type: none"> - последовательности для контроля объектов мультимедиа; - атрибуты интеграции 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - распознаванию форматов графических файлов; - вставке изображений в документы Web с помощью инструментов HTML; - вставке видеофрагментов в документы Web с помощью инструментов HTML; - вставке звуковых фрагментов в документы Web с помощью инструментов HTML; - настройке свойств изображений в документах Web; - интеграции мультимедийных объектов с помощью инструментов HTML. <p><i>Проекты (этап интеграции):</i></p> <ul style="list-style-type: none"> - Вставка мультимедийных объектов в документы Web – составные части некоторого сайта ("История моего города/села", "Птицы Центрального парка", "Статистика погоды" и др.).
<ul style="list-style-type: none"> - Создание таблиц в документах Web. - Редактирование таблиц в документах Web. - Организация контента документов Web с помощью таблиц. 	<p>Организация контента</p> <p>Таблицы:</p> <ul style="list-style-type: none"> - заголовков; - строка; - столбец; - ячейка; - рамка <p>Редактирование таблиц:</p> <ul style="list-style-type: none"> - объединение ячеек; - отступы; - Дизайн ячеек 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - созданию таблиц в документах Web с помощью инструментов HTML; - редактированию таблиц в документах Web с помощью инструментов HTML; - применению таблиц для эффектов дизайна документов Web. <p><i>Проекты (этап корректировки дизайна, презентация):</i></p> <ul style="list-style-type: none"> - Настройка структуры документов Web – составных частей некоторого сайта – с помощью таблиц, созданных инструментами HTML. ("История моего города/села", "Птицы Центрального парка", "Статистика погоды" и др.); - Получение эффектов дизайна для некоторого сайта ("История моего города/села", "Птицы Центрального парка", "Статистика погоды" и др.); - Презентация проектов.

* Только для реального профиля.

** Только для расширения.

По окончании XI класса ученик может:

- применять составные данные для создания моделей информационных объектов, систем и процессов, часто встречающихся в повседневной деятельности;
- выбирать составные типы данных в зависимости от специфики информации из решаемых задач;
- реализовывать на одном из языков высокого уровня алгоритмы обработки составных данных, необходимых для решения задач, связанных с цифровой обработкой информации;
- использовать основные понятия теории информации, компьютерной арифметики и булевой алгебры для создания информационных моделей;
- интерпретировать результаты выполнения разработанных компьютерных программ;
- идентифицировать общую структуру применяемых цифровых систем, принципы работы систем передачи, накопления и обработки информации,

проявляя следующие доминирующие специфические отношения:

- правильность и когерентность в использовании специфической терминологии;
- внимание и интерес к созданию алгоритмов;
- инициативу и настойчивость при создании алгоритмов;
- любопытство и заинтересованность в использовании методов и инструментов, специфичных для цифровой обработки;
- использование критического мышления для принятия соответствующего плана алгоритмизации задач;
- любопытство и креативность в интеграции знаний и умений из области информатики со знаниями и умениями из других областей;
- соблюдение правил безопасности, эргономики, этики и дизайна при создании и распространении компьютерных программ.

Единицы компетенций	Единицы содержания	Рекомендуемая учебная деятельность и школьные продукты
<ul style="list-style-type: none"> - Обработка данных с помощью стандартных подпрограмм и подпрограмм, определенных пользователем. - Организация передачи данных между вызываемой программой/подпрограммой и вызываемой подпрограммой. - Структурное проектирование алгоритма и программы. - Использование рекурсии для решения задач. 	<p style="text-align: center;">1. Подпрограммы</p> <p>Подпрограммы:</p> <ul style="list-style-type: none"> - задачи и подзадачи; - главная и вызываемая программы; - подпрограммы и вызовы подпрограмм; - типы подпрограмм <p>Передача данных между вызываемой программой/подпрограммой и вызываемой подпрограммой:</p> <ul style="list-style-type: none"> - метод передачи управления в случае вызова подпрограммы; - метод передачи аргументов и возврата результатов <p>Области видимости:</p> <ul style="list-style-type: none"> - глобальные и локальные переменные <p>Побочные эффекты*:</p> <ul style="list-style-type: none"> - причины возникновения побочных эффектов; - способ избежать побочных эффектов; - разработка программ, не содержащих побочных эффектов 	<p>Упражнения по:</p> <ul style="list-style-type: none"> - использованию терминов „задача, подзадача, главная программа, вызываемая программа, подпрограмма“; - объяснению способа выполнения вызовов подпрограмм (процедур и функций); - аргументации необходимости разделения комплексных задач на элементарные подзадачи; - определению подпрограмм с помощью металингвистических формул и синтаксических диаграмм; - идентификации подзадач, решение которых требует использования подпрограмм; - наблюдению и пониманию информационных (данных) и управляющих (вызовов) связей между главной программой и вызываемыми подпрограммами; - использованию стандартных функций и процедур языка; - разработке программ, использующих стандартные функции языка; - разработке программ, использующих функции, определенные пользователем; - разработке программ, которые не содержат побочных эффектов. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Способ передачи данных в случае использования параметров-значений и параметров-переменных; - Сложность написания программ с использованием и без использования подпрограмм; - Рекурсия и итерация.

	<p>Рекурсия*:</p> <ul style="list-style-type: none"> - способ передачи управления при выполнении рекурсивных алгоритмов; - способ выделения памяти при выполнении рекурсивных алгоритмов; - достоинства и недостатки рекурсии; - разработка рекурсивных подпрограмм <p>Синтаксис объявлений и вызовов подпрограмм</p>	<p>Проекты:</p> <ul style="list-style-type: none"> - Подпрограммы для вычисления арифметических и геометрических средних заданного набора чисел; - Подпрограммы для элементарного анализа текстов; - Подпрограммы для выполнения арифметических вычислений с длинными числами; - Подпрограммы для сортировки массивов (по возрастанию, убыванию); - Подпрограммы для лексикографической сортировки слов.
2. Техники программирования*		
<ul style="list-style-type: none"> - Оценка сложности алгоритмов. - Использование техник программирования для решения задач из различных областей. - Выбор техники программирования, адекватной данной задаче. 	<p>Анализ алгоритмов:</p> <ul style="list-style-type: none"> - оценивание необходимого объема памяти; - временная сложность алгоритма; - классификация алгоритмов в зависимости от сложности <p>Итеративные и рекурсивные подходы</p> <p>Метод полного перебора:</p> <ul style="list-style-type: none"> - техника полного перебора; - сложность метода полного перебора; - области использования метода полного перебора 	<p>Упражнения по:</p> <ul style="list-style-type: none"> - использованию терминологии из области сложности алгоритмов; - аргументации необходимости анализа сложности алгоритмов; - оценке сложности алгоритмов; - решению задач с помощью часто используемых техник программирования.

	<p>Метод Greedy:</p> <ul style="list-style-type: none"> - техника Greedy; - сложность техники Greedy; - области применения техники Greedy <p>Метод возврата с возвратом**:</p> <ul style="list-style-type: none"> - техника возврата с возвратом; - сложность техники возврата с возвратом; <p>области применения техники возврата с возвратом</p> <p>Метод "разделяй и властвуй"***:</p> <ul style="list-style-type: none"> - техника "разделяй и властвуй"; - сложность техники "разделяй и властвуй"; - области применения техники "разделяй и властвуй" 	<p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Итеративные и рекурсивные подходы при реализации алгоритмов; - Использование полного перебора и техники Greedy для поиска решений; - Метод перехода с возвратом для реализации алгоритмов поиска решений; - Метод "разделяй и властвуй" для реализации алгоритмов поиска решений. <p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Разработка алгоритмов сортировки и оценка их сложности; - Разработка алгоритмов обработки таблиц и оценка их сложности; - Сравнительный анализ итеративных и рекурсивных алгоритмов, предназначенных для создания и обработки динамических структур данных; - Закраска замкнутых областей; - Поиск пути в лабиринте; - Задачи на шахматной доске; - Закраска карт; - Раскройка ткани; - Задача о рюкзаке; - Задача коммивояжера.
3. Моделирование и численные методы*		
<ul style="list-style-type: none"> - Применение критериев классификации моделей. - Разработка математических моделей. - Аргументация необходимости применения моделирования в экономической сфере и социальной жизни. 	<p>Модели и моделирование</p> <p>Классификация моделей:</p> <ul style="list-style-type: none"> - материальные; - идеальные; - математические <p>Математическое моделирование</p>	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - идентификации и классификации моделей; - объяснению понятий модель и моделирование; - перечислению критериев классификации моделей; - выбору характеристик моделируемого объекта в зависимости от назначения модели; - разработке идеальных моделей для физических явлений;

<ul style="list-style-type: none"> - Идентификация аналитических решений и решений, получаемых с помощью вычислительного эксперимента. - Выбор типа решения в зависимости от природы задачи. 		<ul style="list-style-type: none"> - разработке идеальных моделей для плоских и объемных геометрических фигур; - разработке математических моделей для явлений и процессов из области физики, биологии, химии, социальной сферы. <p><i>Задачи по:</i></p> <ul style="list-style-type: none"> - переводу разработанных математических моделей на язык программирования высокого уровня; - исследованию результатов моделирования в зависимости от значений исходных данных.
<ul style="list-style-type: none"> - Идентификация аналитических решений и решений, получаемых с помощью вычислительного эксперимента. - Выбор типа решения в зависимости от природы задачи. 	<p>Аналитические решения и решения, получаемые с помощью вычислительного эксперимента</p>	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - различению аналитических решений и решений, получаемых с помощью вычислительного эксперимента; - идентификации задач, которые могут быть решены аналитическими методами, и задач, которые могут быть решены с помощью вычислительного эксперимента; - объяснению общих методов получения аналитических решений и решений, основанных на вычислительном эксперименте. <p><i>Задачи по:</i></p> <ul style="list-style-type: none"> - контролю эволюции известного процесса путем проведения вычислительного эксперимента; - автоматизации контроля путем разработки программ моделирования на языке высокого уровня (контроль текущих координат робота, контроль уровня жидкости в бассейне и пр.).

<ul style="list-style-type: none"> - Планирование и реализация процесса решения задач на компьютере. 	<p>Этапы решения задач на компьютере:</p> <ul style="list-style-type: none"> - формулировка задачи; - разработка математической модели; - разработка алгоритма; - написание программы; - тестирование программы; - анализ и интерпретация полученных результатов 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - идентификации этапов решения задачи на компьютере; - объяснению взаимодействия между математической моделью и алгоритмом, алгоритмом и программой, программными и аппаратными ресурсами компьютера; - планированию процесса решения задачи на компьютере. <p><i>Задачи по:</i></p> <ul style="list-style-type: none"> - разработке программ и наборов тестов для элементарных алгоритмов (определение элементов с максимальным/минимальным значением в массиве, решение квадратичных уравнений и т.п.). <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Проверка правильности формулирования задачи (например, обработка элементов массива, в котором не указаны ограничения по количеству элементов и т.п.).
<ul style="list-style-type: none"> - Идентификация точных значений и их приближений. - Определение абсолютной и относительной погрешности. - Оценка вычислительных погрешностей, вызванных ошибками входных данных. - Оценка вычислительных погрешностей, вызванных особенностями представления чисел в памяти. 	<p>Погрешности в численных методах:</p> <ul style="list-style-type: none"> - абсолютная погрешность; - относительная погрешность <p>Погрешности:</p> <ul style="list-style-type: none"> - входных данных; - округления; - приближения; - метода; - задачи 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - идентификации точного и вычисленного решения; - вычислению абсолютной и относительной погрешности. <p><i>Задачи по:</i></p> <ul style="list-style-type: none"> - моделированию стандартного (расширенного) арифметического калькулятора на языке программирования высокого уровня (с заданным количеством значащих десятичных знаков результата); - оценке погрешности значений, вычисленных с помощью компьютерных программ; - оценке погрешностей задачи; - оценке погрешностей метода; - оценке погрешностей приближения. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Воздействие погрешностей входных данных на оптимальность вычисляемого решения;

<ul style="list-style-type: none"> - Использование элементарных алгоритмов для отделеия корней уравнения на заданном интервале. - Идентификация условий применения методов половинного деления, хорд, Ньютона. - Разработка на языке программирования высокого уровня программы для итеративного вычисления корня алгебраического или трансцендентного уравнения методами половинного деления, хорд, Ньютона. - Выбор метода для решения алгебраических и трансцендентных уравнений (половинного деления, хорд, Ньютона), наиболее подходящего для конкретной задачи. 	<p>Решение алгебраических и трансцендентных уравнений с помощью компьютера:</p> <ul style="list-style-type: none"> - корень уравнения; - отделеия корней; - метод половинного деления; - метод хорд; - метод Ньютона**; - рекуррентные формулы; - фиксированные концы; - начальное приближение 	<ul style="list-style-type: none"> - Воздействие погрешностей входных данных на корректность вычисляемого решения; - Воздействие погрешности метода на корректность вычисляемого решения; (метод Greedy). <p><i>Проект:</i></p> <ul style="list-style-type: none"> - Разработка на языке программирования высокого уровня программы для сложения (вычитания) больших чисел (до 100 цифр) без аппроксимации результата. 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - отделеию корней методом перебора; - отделеию корней с помощью программ вычерчивания графиков; - описанию алгоритма реализации метода половинного деления (хорд, Ньютона); - разработке программы для итеративного вычисления решений алгебраических и трансцендентных уравнений методом половинного деления, хорд, Ньютона; - проверке условий применимости изучаемых методов для заданных уравнений; - отделеию фиксированного конца для метода хорд; - выбору начального приближения для метода Ньютона. <p><i>Решение задач:</i></p> <ul style="list-style-type: none"> - решение задач, математическая модель которых описывается алгебраическими и трансцендентными уравнениями. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Сравнительный анализ результатов отделеия корней уравнения методом перебора и с помощью приложений для вычерчивания графиков функций; - Сравнительный анализ решений, вычисленных методом половинного деления и методом хорд;
---	--	---	---

4. Базы данных		
<ul style="list-style-type: none"> - Описание структуры иерархических, сетевых и реляционных баз данных. 	<p>Понятия и концепты. Типы баз данных:</p> <ul style="list-style-type: none"> - иерархические; - сетевые; - реляционные 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - объяснению смысла терминов "База данных", "система управления базами данных"; - интуитивному определению (через рисунок) структуры иерархических, сетевых и реляционных баз данных; - различению терминов "данные", "информация", "файл", "база данных", "система управления базами данных".
<ul style="list-style-type: none"> - Описание структуры и функций систем управления базами данных. - Различение этапов разработки базы данных. - Распознавание ролей каждого из участников при разработке и использовании баз данных. 	<p>Системы управления базами данных:</p> <ul style="list-style-type: none"> - структура; - функции <p>Этапы разработки базы данных</p>	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - схематичному представлению (через рисунок) структуры систем управления базами данных; - объяснению назначения каждого объекта реляционной базы данных; - схематичному представлению (через рисунок) потоков данных между объектами реляционной базы данных; - различению этапов разработки базы данных и объяснению содержания каждого этапа; - объяснению ролей каждого из участников в разработке и использовании баз данных. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Этапы проектирования базы данных. <p><i>Проекты (примеры):</i></p> <ul style="list-style-type: none"> - Приложения для создания и управления базами данных; - Описание этапов создания заданной базы данных ("моя школа", "библиотека", "магазин" и т. п.).

<ul style="list-style-type: none"> - Проектирование структуры таблиц базы данных. - Создание таблиц с помощью мастера таблиц или путем самостоятельного проектирования. - Использование способов ввода данных в таблицах. - Создание и редактирование таблиц для задач из области физики, биологии, химии, географии и пр. - Установление корреляций между таблицами. - Использование операций, предназначенных для сортировки записей поиска и замены значений. - Разработка фильтров для отбора записей. 	<p>Таблицы:</p> <ul style="list-style-type: none"> - назначение и структура таблиц; - создание таблиц; - ввод данных в таблицы; - изменение структуры таблиц; - редактирование записей; - сортировка записей; - поиск и замена значений; - создание и использование фильтров; - установка корреляций между таблицами 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - созданию таблиц с помощью мастера таблиц или путем самостоятельного проектирования; - различению типов данных в полях таблицы; - определению свойств полей; - выбору первичного ключа; - объяснению методов ввода данных в таблицы; - вводу данных в созданные таблицы; - изменению структуры таблиц; - редактированию записей; - форматированию данных; - интуитивному объяснению (используя диаграммы корреляции) понятия коррелирующих таблиц; - объяснению свойств каждого типа корреляции; - установке корреляций между таблицами; - сортировке записей, поиска и замены значений; - разработке фильтров. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Общие понятия по нормализации таблиц; - Проверка правильности данных в полях таблицы; - Использование масок ввода (шаблонов) для ввода данных; - Обеспечение целостности данных; - Каскадное редактирование данных. <p><i>Проекты (примеры):</i></p> <ul style="list-style-type: none"> - Создание таблиц для базы данных ("Моя школа", "Библиотека", "Магазин" и т. п.); - Ввод данных в таблицы для базы данных ("Моя школа", "Библиотека", "Магазин" и т. п.); - Редактирование данных в таблицах базы данных ("Моя школа", "Библиотека", "Магазин" и т. п.); - Установка корреляций между таблицами базы данных ("Моя школа", "Библиотека", "Магазин" и т. п.).
---	---	---

<ul style="list-style-type: none"> - Выбор адекватных типов запросов для обработки данных. - Создание запросов с помощью мастера запросов или путем самостоятельного проектирования. - Создание запросов для отбора данных. 	<p>Запросы:</p> <ul style="list-style-type: none"> - назначение и структура запросов; - создание запросов с помощью мастера запросов; - проектирование запросов*; - сортировка и группирование запросов* 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - интуитивному определению (через рисунок) понятий "запрос" и "динамический набор результатов запроса"; - созданию стандартных запросов с помощью мастера запросов; - созданию простых и комплексных запросов; - созданию запросов. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Структурированные языки запросов. <p><i>Проекты (примеры):</i></p> <ul style="list-style-type: none"> - Формулирование заявок на запросы для разрабатываемой базы данных ("Моя школа", "Библиотека", "Магазин и т. п.); - Создание запросов для базы данных ("Моя школа", "Библиотека", "Магазин и т. п.); - Сортировка и группировка записей базы данных ("Моя школа", "Библиотека", "Магазин и т. п.).
<ul style="list-style-type: none"> - Идентификация компонентов и использование правил формирования выражений. - Разработка запросов действия. - Использование техник группирования и подведения итогов. 	<p>Выражения*:</p> <ul style="list-style-type: none"> - составление выражений; - запросы на добавление, актуализацию и исключение; - группировка и подведение итогов в запросе 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - написанию и проверке выражений; - составлению выражений с помощью системы подсказки; - разработке и редактированию запросов действия; - группировке и подведению итогов по данным из запросов, предложенных преподавателем или разработанных учащимися; - проверке выражений. <p><i>Проекты (примеры):</i></p> <ul style="list-style-type: none"> - Формулирование заявок на запросы на базе выражений и их тестирование для разрабатываемой базы данных ("Моя школа", "Библиотека", "Магазин и т. п.); - Добавление, актуализация и удаление записей для разрабатываемых баз данных ("Моя школа", "Библиотека", "Магазин и т. п.); - Группировка и подведение итогов по данным для разрабатываемых баз данных ("Моя школа", "Библиотека", "Магазин и т. п.).

<ul style="list-style-type: none"> - Идентификация составных элементов формуляра. - Разработка формуляра с помощью мастера формуляров * или путем самостоятельного проектирования. - Применение технологии модификации формуляров. - Использование формуляров для просмотра, редактирования и проверки правильности данных. - Разработка формуляров, основывающихся на коррелирующих таблицах. 	<p>Формуляры:</p> <ul style="list-style-type: none"> - назначение и структура формуляра; - использование формуляров; - создание формуляров с помощью мастера формуляров; - проектирование формуляров*; - модификация формуляров*; - формуляры, основанные на коррелированных таблицах*; - форматирование данных* 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - интуитивному представлению (через рисунок) структуры формуляров и потоков данных между формулярами и остальными объектами базы данных; - созданию стандартных формуляров с помощью мастера формуляров* или путем самостоятельного проектирования; - определению свойств формуляров, их элементов контроля и разделов; - редактированию формуляров; - адаптации формуляров; - использованию формуляров; - объяснению принципа функционирования формуляров на базе коррелированных таблиц; - форматированию данных в базах данных; - проектированию запросов на основе коррелированных таблиц; - созданию и использованию вложенных формуляров. <p><i>Проекты (примеры):</i></p> <ul style="list-style-type: none"> - Создание формуляров на основании таблицы базы данных ("Моя школа", "Библиотека", "Магазин и т. п.); - Использование и редактирование формуляров базы данных ("Моя школа", "Библиотека", "Магазин и т. п.); - Создание формуляров на коррелирующих таблицах базы данных ("Моя школа", "Библиотека", "Магазин и т. п.); - Форматированию данных в таблицах базы данных ("Моя школа", "Библиотека", "Магазин и т. п.).
---	---	---

<ul style="list-style-type: none"> - Идентификация составных компонентов отчета. - Разработка отчетов и подотчетов с помощью подсказки или путем самостоятельного проектирования. - Использование методов модификации отчетов. - Использование методов группировки данных в отчете. 	<p>Отчёты:</p> <ul style="list-style-type: none"> - назначение и структура отчетов; - создание отчетов с помощью мастера отчетов; - проектирование отчетов*; - создание отчетов на базе коррелированных таблиц*; - группировка и подведение итогов по данным в отчете* 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - интуитивному представлению (через рисунок) структуры отчетов и потоков данных между отчетами и остальными объектами базы данных; - разработке отчетов с помощью мастера отчетов или путем самостоятельного проектирования; - анализу структуры и особенностей отчета; - созданию и использованию отчетов и подотчетов на базе данных, предложенных учителем или созданных учениками; - группировке и подведению итогов по данным из отчетов; <p><i>Проекты (примеры):</i></p> <ul style="list-style-type: none"> - Создание отчетов на базе таблицы базы данных ("Моя школа", "Библиотека", "Магазин и т. п.); - Создание отчетов на базе коррелирующих таблиц базы данных ("Моя школа", "Библиотека", "Магазин и т. п.); - Группировка и подведение итогов по данным в отчете для разрабатываемых баз данных ("Моя школа", "Библиотека", "Магазин и т. п.).
<ul style="list-style-type: none"> - Техническое обслуживание баз данных. 	<p>Сжатие и восстановление баз данных* Безопасность баз данных* Администрирование баз данных*</p>	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - сжатию и восстановлению базы данных; - созданию паролей доступа для различных категорий пользователей базы данных. <p><i>Проекты (примеры):</i></p> <ul style="list-style-type: none"> - Сжатие и восстановление базы данных ("Моя школа", "Библиотека", "Магазин и т. п.); - Создание паролей доступа для различных категорий пользователей базы данных ("Моя школа", "Родное село", "Магазин" и т. п.).

5-А. Углублённая обработка информации в базах данных

<ul style="list-style-type: none"> - Использование средств сортировки записей по сложным критериям. 	<p>Сортировка записей:</p> <ul style="list-style-type: none"> - частичная сортировка списков; - полная сортировка списков; - сортировка по нескольким ключам 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - создание баз данных в форме списков; - частичной сортировке списков; - полной сортировке списков; - сортировке по нескольким ключам. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Преимущества и недостатки баз данных в форме списков. <p><i>Проекты (примеры):</i></p> <ul style="list-style-type: none"> - Разработка баз данных, содержащих информацию о книгах из личной библиотеки, аудиозаписях персональной фонотеки, видеозаписях из личного архива. Сортировка записей в разработанных базах данных; - Оценка возможности создания базы данных в виде списка для библиотеки школы, в которой учится ученик.
<ul style="list-style-type: none"> - Использование средств отбора записей по сложным критериям. 	<p>Отбор записей из базы данных в форме списка:</p> <ul style="list-style-type: none"> - отбор с помощью встроенных фильтров; - отбор с помощью фильтров, разработанных пользователем; - отбор с помощью сложных условий; - копирование отобранных записей 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - отбору записей при помощи встроенных фильтров; - отбору записей с помощью фильтров, разработанных пользователем; - отбору записей с помощью сложных условий. <p><i>Тематическое исследование:</i></p> <ul style="list-style-type: none"> - Сравнительный анализ преимуществ и недостатков встроенных фильтров и фильтров, разработанных пользователем. <p><i>Проекты (примеры):</i></p> <ul style="list-style-type: none"> - Создание фильтров для отбора записей в базах данных, содержащих информацию о книгах из личной библиотеки, аудиозаписях персональной фонотеки, видеозаписях из личного архива; - Оценка возможностей использования фильтров для отбора записей из базы данных в виде списка для библиотеки школы, в которой учится ученик.

<ul style="list-style-type: none"> - Использование средств подведения итогов в базах данных в виде списка. 	<p>Подведение итогов в базах данных в виде списка:</p> <ul style="list-style-type: none"> - итоговые функции; - частичные и общие итоги; - условия для подведения итогов; - консолидация по положению; - консолидация по категориям 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - операциям по подведению итогов; - объяснению операций консолидации; - подведению частичных итогов; - подведению общих итогов; - консолидации данных по положению; - консолидации данных по категориям. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Сравнительный анализ достоинств и недостатков методов подведения итогов; - Сравнительный анализ достоинств и недостатков методов консолидации данных. <p><i>Проекты (примеры):</i></p> <ul style="list-style-type: none"> - Подведение итогов в базах данных, содержащих информацию о: книгах из личной библиотеки, аудиозаписях персональной фонотеки, видеозаписях из личного архива; - Консолидация данных в базах данных, содержащих информацию о: книгах из личной библиотеки, аудиозаписях персональной фонотеки, видеозаписях из личного архива; - Оценка возможностей использования средств подведения итогов для расширенной обработки данных базы данных в виде списка для библиотеки школы, в которой учится ученик.
<ul style="list-style-type: none"> - Использование синтетических таблиц для расширенной обработки информации из баз данных в виде списков. 	<p>Создание и использование синтетических таблиц (pivot):</p> <ul style="list-style-type: none"> - область интересующих значений; - заголовки строк; 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - объяснению алгоритмов обработки данных, содержащихся в синтетических таблицах; - объяснению способа работы синтетических таблиц; - созданию синтетических таблиц.

	<ul style="list-style-type: none"> - заголовки столбцов; - итоговые функции; - уровни детализации 	<p><i>Тематическое исследование:</i></p> <ul style="list-style-type: none"> - Сравнительный анализ достоинств и недостатков методов обработки данных с применением синтетических таблиц. <p><i>Проекты (примеры):</i></p> <ul style="list-style-type: none"> - Создание синтетических таблиц для обработки данных в базах данных, содержащих информацию о: книгах из личной библиотеки, аудиозаписях персональной фонотеки, видеозаписях из личного архива; - Оценка возможностей использования синтетических таблиц для расширенной обработки данных базы данных в виде списка для библиотеки школы, в которой учится ученик.
5-В. Экспериментальные методы в гуманитарных науках		
<ul style="list-style-type: none"> - Определение переменных, используемых в гуманитарных науках. 	<p>Экспериментальная методология гуманитарных наук. Понятие переменной:</p> <ul style="list-style-type: none"> - определение; - понятие причинности; - понятие взаимодействия; - понятие корреляции 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - объяснению понятий переменной, причинности, взаимодействия, корреляции; - идентификации некоторых социальных феноменов и набора переменных, необходимых для их анализа; - определению часто встречающихся в гуманитарных науках переменных. <p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Разработка набора переменных, которые могут быть использованы при опросе среди учеников учебного заведения: уровень удовлетворённости физическими условиями в школе, степень участия учащихся в общественной деятельности, отношение учащихся к любым случаям копирования; - Разработка набора переменных, которые могут быть использованы при опросе в случае местных выборов в населённом пункте, в котором расположено учебное заведение;

<p>- Определение контрольных групп.</p>		<p>- Разработка набора переменных, которые могут быть использованы при опросе в населённом пункте, в котором находится учебное заведение: качество дорог, состояние окружающей среды, возможное место постройки развлекательного центра, намерения граждан внести свой финансовый вклад или безвозмездный труд на благоустройство родного села и т. д.</p>
<p>- Определение контрольных групп.</p>	<p>Контрольные группы и паразитные переменные:</p> <ul style="list-style-type: none"> - понятие контрольной группы; - синхронные опыты; - диахронные опыты; - опыты в контрольных группах с артефактом; - паразитные переменные 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - объяснению понятий контрольной группы, синхронного опыта, диахронного опыта, контрольной группы с артефактом, паразитной переменной; - определению типов экспериментов, предложенных учителем; - идентификации контрольных групп и типа эксперимента в зависимости от специфики изучаемого социального феномена. <p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Проектирование контрольных групп и типа эксперимента, которые могут быть использованы при изучении социального поведения учащихся в рамках учебного заведения; - Проектирование контрольных групп и типа эксперимента, которые могут быть использованы при изучении социального поведения избирателей в населённом пункте, где расположено учебное заведение; - Проектирование контрольных групп и типа эксперимента, которые могут быть использованы при изучении отношения жителей села, где находится школа, к качеству дорог, состоянию окружающей среды, потенциальному месту строительства развлекательного центра, намерениям граждан внести свой финансовый вклад или безвозмездный труд на благоустройство родного села и т. д.

<ul style="list-style-type: none"> - Разработка плана эксперимента и выбор субъектов. 	<p>План эксперимента и выбор субъектов:</p> <ul style="list-style-type: none"> - однофакторный план; - многофакторный план; - выбор субъектов 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - объяснению понятий "однофакторный и многофакторный план"; - определению типов планов, предложенных учителем; - выбору субъектов в зависимости от специфики изучаемого социального феномена. <p><i>Тематическое исследование:</i></p> <ul style="list-style-type: none"> - Сравнительный анализ достоинств и недостатков одно- и многофакторных планов. <p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Разработка планов эксперимента и выбор субъектов в случае изучения социального поведения учащихся в рамках учебного заведения; - Разработка планов эксперимента и выбор субъектов в случае изучения социального поведения избирателей населённого пункта, где расположена школа; - Разработка планов эксперимента и выбор субъектов в случае изучения отношения жителей населённого пункта, где находится школа, к качеству дорог, состоянию окружающей среды, возможному месту строительства развлекательного центра, намерению граждан внести свой финансовый вклад или безвозмездный труд на благоустройство родного села и т. д.
<ul style="list-style-type: none"> - Выбор шкалы измерения. 	<p>Математическое описание первичной информации из области гуманитарных наук</p> <p>Шкалы измерения и их графическое представление:</p> <ul style="list-style-type: none"> - номинальные шкалы, - упорядоченные шкалы, - интервальные шкалы, - шкалы отношений <p>Выбор шкалы измерения</p>	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - объяснению понятий "шкала", "номинальная шкала", "упорядоченная шкала", "интервальная шкала", "шкала отношений"; - определению типов шкал, предложенных учителем; - выбору шкал в зависимости от специфики изучаемого социального феномена. <p><i>Тематическое исследование:</i></p> <ul style="list-style-type: none"> - Сравнительный анализ достоинств и недостатков шкал измерения.

		<p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Выбор шкал измерения, которые могут быть использованы при проведении опроса среди учащихся в рамках учебного заведения: уровень удовлетворённости физическими условиями в школе, степень участия учащихся во внешкольных мероприятиях, отношение учеников к любым случаям копирования; - Выбор шкал измерения, которые могут быть использованы в случае местных выборов в населённом пункте, в котором располагается учебное заведение; - Выбор шкал измерения, которые могут быть использованы при проведении опроса граждан в населённом пункте, где расположено учебное заведение: качество дорог, состояние окружающей среды, возможное место строительства развлекательного центра, намерение граждан внести свой финансовый вклад или безвозмездный труд на благоустройство родного села и т. д.
<ul style="list-style-type: none"> - Числовое описание данных, используемых для изучения феноменов из области гуманитарных наук. 	<p>Итоги и числовое описание данных, используемых для изучения феноменов из области гуманитарных наук:</p> <ul style="list-style-type: none"> - индикаторы центральной тенденции и дисперсии в случае интервальных шкал; - индикаторы центральной тенденции и дисперсии в случае упорядоченных шкал; - индикаторы центральной тенденции и дисперсии в случае номинальных шкал <p>Выбор индикаторов в зависимости от типа переменных</p>	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - объяснению индикаторов центральной тенденции и дисперсии для каждого типа шкалы; - интерпретации числовых значений индикаторов, предложенных учителем; - выбор индикаторов в зависимости от специфики изучаемого социального феномена. <p><i>Тематическое исследование:</i></p> <ul style="list-style-type: none"> - Сравнительный анализ достоинств и недостатков индикаторов центральной тенденции и дисперсии для каждого типа шкалы (интервальной, упорядоченной, номинальной).

<p>- Получение выборки.</p>	<p>Понятие населения и выборки:</p> <ul style="list-style-type: none"> - определены; - методы выборки (эмпирические, квотные, типовые, вероятностные, случайные, стратифицированные) 	<p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Выбор индикаторов, которые могли бы быть использованы при изучении результатов опроса среди учащихся в рамках учебного заведения: уровень удовлетворённости физическими условиями в школе, степень участия учащихся во внешкольных мероприятиях, отношение учеников к любым случаям копирования; - Выбор индикаторов, которые могли бы быть использованы при изучении результатов местных выборов, в населённом пункте, где расположено учебное заведение; - Выбор индикаторов, которые могли бы быть использованы при изучении результатов проведения опроса граждан в населённом пункте, где расположено учебное заведение: качество дорог, состояние окружающей среды, возможное место строительства развлекательного центра, намерение граждан внести свой финансовый вклад или безвозмездный труд на благоустройство родного села и т. д.
<p>-</p>	<p>Упражнения по:</p> <ul style="list-style-type: none"> - объяснению понятий "население", "индивид", "характеристика", "выборка"; - взятию выборки следующими методами: эмпирическим, квотным, типовым, вероятностным, случайным, стратифицированным. <p><i>Тематическое исследование:</i></p> <ul style="list-style-type: none"> - Сравнительный анализ достоинств и недостатков методов получения выборки (эмпирических квотных, типовых, вероятностных, случайных, стратифицированных). <p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Проектирование выборки для проведения опроса среди учащихся в рамках учебного заведения: уровень удовлетворённости физическими условиями в школе, степень участия учащихся во внешкольных мероприятиях, отношение учеников к любым случаям копирования; 	

<ul style="list-style-type: none"> - Использование электронных таблиц для анализа данных в гуманитарных науках. 		<ul style="list-style-type: none"> - Проектирование выборки для проведения опроса избирателей относительно намерений голосовать в населённом пункте, где расположена школа; - Проектирование выборки для проведения опроса граждан в населённом пункте, где расположено учебное заведение: качество дорог, состояние окружающей среды, возможное место строительства развлекательного центра, намерение граждан внести свой финансовый вклад или безвозмездный труд на благоустройство родного села и т. д.
<ul style="list-style-type: none"> - Использование электронных таблиц для анализа данных в гуманитарных науках. 	<p>Использование электронных таблиц для углублённого анализа данных в гуманитарных науках:</p> <ul style="list-style-type: none"> - сбор данных; - проверка данных; - систематизация данных, - группировка данных; - вычисление индикаторов центральной тенденции; - создание гистограмм; - интерпретация полученных результатов 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - систематизации возможностей, предлагаемых электронными таблицами для обработки данных из области гуманитарных наук. <p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Анализ данных, полученных в рамках опроса, реализованного среди учеников в рамках учебного заведения: уровень удовлетворённости физическими условиями в школе, степень участия учащихся во внешкольных мероприятиях, отношение учеников к любым случаям копирования; - Анализ данных, полученных в рамках опроса, реализованного среди граждан населённого пункта, где расположено учебное заведение, относительно их намерений голосовать на местных выборах; - Анализ данных, полученных в рамках опроса, проведенного в населённом пункте, где расположено учебное заведение, относительно: качества дорог, состояния окружающей среды, возможного места строительства развлекательного центра, намерений граждан внести свой финансовый вклад или безвозмездный труд на благоустройство родного села и т. д.

<p>- Использование специализированных программ для выполнения социальных исследований.</p>	<p>Программные продукты для социальных наук:</p> <ul style="list-style-type: none"> - графический интерфейс; - описание переменных; - импорт данных; - проверка данных; - создание таблиц частот; - создание таблиц непредвиденных обстоятельств; - создание таблиц ассоциаций; - создание графиков по количественным и качественным переменным; - вычисление индикаторов центральной тенденции; - интерпретация полученных результатов 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - исследованию возможностей, предлагаемых программными продуктами для социальных наук при обработке данных; - формированию индивидуальных способностей работы с программными продуктами для социальных наук. <p><i>Тематическое исследование:</i></p> <ul style="list-style-type: none"> - Сравнительный анализ достоинств и недостатков вычислительных приложений и программных продуктов для социальных наук при углублённой обработке данных в области гуманитарных наук. <p><i>Проекты:</i></p> <ul style="list-style-type: none"> - Анализ данных, полученных в рамках опроса, проведенного среди учеников в рамках учебного заведения: уровень удовлетворённости физическими условиями в школе, степень участия учащихся во внешкольных мероприятиях, отношение учеников к любым случаям копирования; - Анализ данных, полученных в рамках опроса, реализованного среди граждан населённого пункта, где расположено учебное заведение, относительно их намерений голосовать на местных выборах. - Анализ данных, полученных в рамках опроса, реализованного среди граждан населённого пункта, где расположено учебное заведение, относительно качества дорог, состояния окружающей среды, предполагаемого строительства Центра отдыха, намерений граждан вносить денежные средства или безвозмездно работать по благоустройству местности и др.
--	---	---

5-С. Веб-программирование		
<ul style="list-style-type: none"> - Использование понятий, связанных с интерактивными Веб-ресурсами. - Нахождение последовательностей кода в Веб-документах. - Распознавание ситуаций, требующих использование интерактивных инструментов на Веб -страницах. 	<p>Интерактивные Веб-документы:</p> <ul style="list-style-type: none"> - основные характеристики; - разметка последовательностей команд в Веб-документах; - вставка последовательности команд 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - распознаванию команд/последовательности команд в тексте Веб-документа; - наблюдению эффектов, оказываемых командами/последовательностями команд на вид Веб-страниц в программах навигации; - добавлению команд/последовательности команд в документы Веб; - просмотру документов Веб, содержащих команды/последовательности команд.
<ul style="list-style-type: none"> - Идентификация первичных инструментов Веб-программирования. - Применение синтаксических правил языка Веб-программирования. - Объявление простых данных. - Объявление составных данных. - Узнавание операторов и выражений языка Веб-программирования. - Использование данных в последовательности команд. 	<p>Первичные элементы некоторого языка Веб-программирования:</p> <ul style="list-style-type: none"> - правила синтаксиса; - константы; - переменные; - операторы; - выражения <p>Простые типы данных:</p> <ul style="list-style-type: none"> - числа; - символы <p>Операторы:</p> <ul style="list-style-type: none"> - арифметические; - реляционные; - логические <p>Структуры данных:</p> <ul style="list-style-type: none"> - строки символов; - массивы; - объекты 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - идентификации констант, переменных, операторов и выражений в последовательностях команд, вставленных в текст Веб-документа; - объявлению переменных согласно существующим типам; - написанию выражений, содержащих константы, переменные, операторы. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Скрипты – способы интеграции последовательностей кода, интерпретируемых в Веб-документах; - Языки программирования, позволяющие создавать скрипты; - Скрипты и безопасность Веб-документов. <p><i>Веб-проекты (проектирование структуры):</i></p> <ul style="list-style-type: none"> - Универсальный математический калькулятор.; - Ассистент питания; - Конфигуратор компьютера.

<ul style="list-style-type: none"> - Уточнение моделей вывода и считывания данных. - Использование инструментов языка для ввода и считывания данных. - Разработка последовательностей кода для ввода и вывода данных в Веб-документах. 	<p>Операции ввода/вывода/контроля Вывод данных:</p> <ul style="list-style-type: none"> - модели вывода; - функции вывода; - параметры вывода; - вывод сообщений; - вывод данных <p>Ввод данных:</p> <ul style="list-style-type: none"> - модели считывания; - функции считывания; - параметры считывания; - считывание простых данных <p>Контроль:</p> <ul style="list-style-type: none"> - контроль действий клавиатуры; - контроль действий мыши 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - распознаванию команд ввода/вывода языка Веб-программирования с помощью скриптов, интегрированных в Веб-документы; - вводу и выводу сообщений с помощью последовательностей кода, входящих в активные Веб-документы; - интерпретации введенных числовых и символьных данных; - выводу результатов вычисления выражений, описанных в последовательностях кода, встроенных в Веб-документы. <p><i>Тематическое исследование:</i></p> <ul style="list-style-type: none"> - Контроль видимости вспомогательных окон, генерируемых в Веб-документе; - Веб-проекты (разработка модулей ввода/вывода данных): - Универсальный математический калькулятор; - Ассистент питания; - Конфигуратор компьютера.
<ul style="list-style-type: none"> - Распознавание ситуаций, в которых необходимо использовать структуры контроля Веб-программирования. - Выбор подходящей структуры контроля. - Реализация структур контроля в виде последовательностей кода, интегрированных в веб-документы. - Тестирование структур контроля из последовательностей кода. 	<p>Структуры контроля.</p> <p>Выбор:</p> <ul style="list-style-type: none"> - элементарный выбор; - множественный выбор <p>Выбор в зависимости от событий:</p> <ul style="list-style-type: none"> - действия с клавиатуры; - действия мышью <p>Повторение (Циклы):</p> <ul style="list-style-type: none"> - со счетчиком; - с предусловием; - с постусловием 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - проектированию в веб-документах действий с использованием команд простого/множественного выбора; - проектированию в веб-документах действий с использованием команд повторения; - правильному составлению команд простого/множественного выбора; - правильному составлению команд повторения; - решению примеров с использованием команд простого/множественного выбора; - решению примеров с использованием команд повторения; - интеграции команд выбора в последовательности кода для управления контентом Веб-документов; - интеграции команд повторения в последовательности кода для управления контентом Веб-документов.

<ul style="list-style-type: none"> - Спецификация функций (подпрограмм) в исходном коде. - Преобразование структуры исходного кода для его последующего разбиения на функции. - Программирование собственных функций. 		<p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Источники уязвимости в Веб-документах; - Безопасность Веб-документов; - Веб-проекты (развитие модулей выбора/повторения действий в проектах) ; - Универсальный математический калькулятор; - Ассистент питания; - Конфигуратор компьютера.
<ul style="list-style-type: none"> - Функции: <ul style="list-style-type: none"> - определение; - выполнение; - возврат результата; - импортированные функции 	<p><i>Упражнения по:</i></p> <ul style="list-style-type: none"> - проектированию в веб-документах действий с использованием функций; - разбиению решаемых в последовательностях кода задач на элементарные подзадачи; - решению подзадач путем написания и реализации функций; - тестированию решенных подзадач; - созданию функций со специфическими свойствами/структурой. <p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Внешние функции – обеспечение безопасности кода; - Разработка дизайна веб-страницы с помощью скриптов. <p><i>Веб-проекты (совершенствование всех модулей путем разбиения решаемых задач на подзадачи, применение функций):</i></p> <ul style="list-style-type: none"> - Универсальный математический калькулятор; - Ассистент питания; - Конфигуратор компьютера. 	<p><i>Тематические исследования:</i></p> <ul style="list-style-type: none"> - Источники уязвимости в Веб-документах; - Безопасность Веб-документов; - Веб-проекты (развитие модулей выбора/повторения действий в проектах) ; - Универсальный математический калькулятор; - Ассистент питания; - Конфигуратор компьютера.

5-D. Динамические структуры данных		
<ul style="list-style-type: none"> - Идентификация задач, для решения которых нужны динамические структуры данных. - Применение динамических структур данных для решения задач из повседневной деятельности. - Разработка и управление часто используемых динамических структур. - Управление внутренней памятью компьютера. 	<p>Динамические переменные</p> <p>Ссылочный тип:</p> <ul style="list-style-type: none"> - статические переменные; - динамические переменные; - ссылки на динамические переменные; - ссылочный тип; - статическое и динамическое выделение памяти; - создание и уничтожение динамических переменных <p>Структуры данных:</p> <ul style="list-style-type: none"> - неявные и явные структуры; - статические и динамические структуры; - однородные и неоднородные структуры; - рекурсивные структуры <p>Динамические структуры данных:</p> <ul style="list-style-type: none"> - односвязные списки, стеки и очереди; - создание изучаемых структур; - обработка изучаемых структур; - извлечение данных из изучаемых структур <p>Двоичные деревья:</p> <ul style="list-style-type: none"> - создание двоичных деревьев; - обход двоичных деревьев; - изменение структуры двоичных деревьев 	<p>Упражнения по:</p> <ul style="list-style-type: none"> - интуитивному представлению (через рисунок) методов динамического распределения памяти; - обоснованию необходимости использования динамических структур данных; - выделению различий между неявными и явными структурами, между однородными и неоднородными структурами, между статическими и динамическими структурами данных; - выбору задач, для решения которых нужны изучаемые структуры данных; - созданию, применению и уничтожению динамических переменных; - разработке программ с использованием динамических переменных; - объяснению способа размещения в оперативной памяти статических и динамических переменных; - хранению и обработке данных с использованием списков, стеков, очередей и двоичных деревьев. <p>Тематические исследования **::</p> <ul style="list-style-type: none"> - Поиск информации в списках, стеках, очередях и двоичных деревьях; - Обход списков, стеков, очередей и двоичных деревьев; - Добавление и удаление данных в списках, стеках, очередях и двоичных деревьях; - Области применения динамических структур данных. <p>Проекты **::</p> <ul style="list-style-type: none"> - Обработка списков абитуриентов лицей; - Обработка списков различных между собой слов, встречающихся в некотором тексте;

	<p>Тип данных указатель (pointer)**:</p> <ul style="list-style-type: none"> - множество значений типа pointer; - множество операций типа pointer; - управление внутренней памятью компьютера 	<ul style="list-style-type: none"> - Визуализация входящего и исходящего потока вагонов в случае железнодорожного депо; - Обработка списка сотрудников некоторого предприятия; - Синтаксический анализ арифметических выражений; - Просмотр очереди на ожидание самолётов, запросивших посадку в некотором аэропорту; - Создание и обработка двоичных деревьев, возникающих в случае турниров "на выбывание"; - Вычисление арифметических выражений, представленных двоичными деревьями.
--	---	--

* Только для реального профиля.

**Расширение.

По окончании XII класса ученик может:

- использовать вспомогательные алгоритмы в процессе создания алгоритмов;
- реализовывать вспомогательные алгоритмы на некотором языке высокого уровня;
- оценивать сложность алгоритмов и компьютерных программ;
- выбирать техники программирования в зависимости от специфики решаемых задач;
- реализовывать наиболее распространённые техники программирования при решении задач;
- создавать и реализовывать на компьютере информационные модели объектов, систем и процессов, часто встречающихся в повседневной деятельности;
- применять численные методы вычислений;
- организовывать и обрабатывать информацию с помощью систем управления базами данных, *демонстрируя следующие преобладающие специфические отношения:*
- правильность и когерентность в использовании специфической терминологии из области цифровых средств;
- инициативу и настойчивость в алгоритмизации задач и реализации алгоритмов;
- любопытство и интерес в освоении методов и инструментов компьютерного моделирования;
- критическое и креативное отношение в процессе познания мира с помощью компьютерного моделирования;
- желание познать себя и мир цифровыми средствами;
- соблюдение правил безопасности, эргономики, этики и дизайна при создании и распространении цифровых продуктов.

V. Методические рекомендации для преподавания – учения – оценивания

Методология обучения – учения информатики основывается на следующих принципах:

- сочетание процессов изучения теоретических знаний с практической деятельностью на компьютере;
- адаптация преподаваемых знаний возрасту учащихся и их предыдущему опыту в области цифровых технологий;
- междисциплинарности;
- соответствие методов преподавания и обучения со спецификой электронного обучения;
- балансировка информационной нагрузки и обеспечение преемственности между классами путем дозирования теоретического материала, с учетом специфики возраста учащихся и в соответствии с техническими характеристиками программ электронного обучения, прикладных программ и операционных систем;
- дифференциация и индивидуализация обучения и учения;
- создание потенциала для продвижения в освоении нового оборудования и программных продуктов, в применении современных информационных технологий.

Основными составляющими технологии обучения – учения информатики являются:

- Использование активных методов, ориентированных на ученика, с использованием возможностей, предоставляемых обучением с помощью компьютера.
- Разумное сочетание теоретического обучения и индивидуальной работы на компьютере.
- Систематическое развитие специфических компетенций дисциплины.

Куррикулум для лицеев по дисциплине *Информатика* служит основой для разработки и организации учебного процесса, направленного на формирование и развитие специфических компетенций дисциплины. Такой подход предусматривает ориентацию обучения информатике на формирование у учащихся способности интегрироваться в информационное общество.

Будучи набором способностей, знаний и навыков, компетенция может быть развита только путем комплексного развития ее доминирующих аспектов:

- *когнитивный аспект*, направленный на использование теорий и понятий из области информатики;
- *функциональный аспект*, который заключается в способности человека работать в определённой области – профессиональной, образовательной, социальной – с использованием цифровых средств;
- *этический аспект*, который относится к личным и социальным ценностям.

В усвоении сообщаемой информации задействованы психические процессы восприятия, запоминания и мыслительные операции. Следовательно, для разработки дидактических заданий будет использована в первую очередь таксономия Блума-Андерсена (Bloom-Anderson) и Блума-Кротволя (Bloom-Krathwohl), ориентированная на достижение обучаемым интеллектуального уровня, заданного стандартами.

С целью усвоения учащимися знаний рекомендуется использование методов: СИНЕЛГ (SINELG), интервью, чтение предлагаемых работ, практические упражнения на компьютере, выполнение задач по развитию алгоритмического мышления.

Функциональный аспект компетенции направлен на развитие интеллектуальных и психомоторных способностей учащихся. Для развития интеллектуального потенциала могут быть использованы таксономии: Симпсона, Дейва.

Методами, рекомендуемыми для дисциплины *Информатика*, являются: изложение теоретического материала, работа на компьютере (индивидуально или под руководством преподавателя), упражнения, решение задач, практическая работа, лабораторные работы, тематические исследования, разработка индивидуальных и/или групповых проектов. Особое внимание необходимо уделить разработке учениками транс- и междисциплинарных проектов, что обеспечит дидактическую реализацию подходов STEM/STEAM¹.

В процессе формирования, углубления и консолидации отношений и ценностей рекомендуется использовать таксономию Кротволя (Krathwohl).

Ориентация процесса преподавания – обучения – оценивания на приобретение компетенций генерирует систему непрерывного оценивания, реализуемую посредством текущего и суммативного (итогового) оценивания.

На основании оценочной деятельности реализуется мотивация учащихся и получение непрерывной обратной связи, которая позволяет оперативно корректировать процесс обучения; стимулировать самооценивание и взаимооценивание; подчеркивать успехи, внедрять выборочное и индивидуальное оценивание.

Инновационным элементом оценивания является возможность использования цифровых образовательных ресурсов для оценивания с помощью компьютера как локально, так и в сети.

В данном контексте ценность текущего оценивания состоит в постоянном, непрерывном формировании у учащихся компетенций, отражённых в образовательных стандартах.

Задания для текущего (формативного) оценивания необходимо разделять по уровням сложности, с целью индивидуализации оценивания и дополнительной мотивации учеников. Разработка заданий для оценивания должна осуществляться в контексте соответствующих таксономий.

¹ STEAM – это сокращение от английских слов: Science, Technology, Engineering, Arts and Mathematics (Наука, Техника, Инженерия, Искусство и Математика).

Методы, используемые для текущего оценивания, предусматривают письменный или устный опрос, а также интерактивные методы: тематические исследования, практические работы, проекты, интерактивные тесты с использованием компьютера.

Практическая деятельность учащихся будет более эффективна в случае их информирования преподавателем относительно тематики работ, способа оценивания (балл/шкала/критерии), условия выполнения.

Реализация непрерывного процесса оценивания позволяет объективно оценивать знания и компетенции учащихся, а также их прогресс в обучении.

Итоговое оценивание может проводиться в конце фазы обучения, в конце изучения Единицы обучения, тем, глав, на тезисах, путем разработки всеобъемлющих тем, охватывающих данную тематическую область.

В качестве составных компонентов инструмента оценивания рекомендуется использование тестовых заданий типа решения задач на компьютере, компьютерных тестов и цифровых продуктов, выполненных учениками.

**GHID
DE IMPLEMENTARE
A CURRICULUMULUI
DISCIPLINAR**

Введение

С 1985 года в учебных заведениях нашей страны началось преподавание нового предмета – *Основ информатики и вычислительной техники*, которые впоследствии превратились в школьную дисциплину сегодняшнего дня – *Информатику*.

В отличие от классических школьных предметов, преподаваемых веками, которые напрямую не подвержены влиянию технологических факторов, *Информатика* находится под постоянным воздействием как быстрых изменений в области информационных технологий, так и постоянного расширения круга лиц, имеющего доступ к этим технологиям.

Вследствие этого, методологические основы и парадигмы разработки куррикулума для этой школьной дисциплины требуют частых изменений, а грань между фундаментальными знаниями (теория информации, теория программируемых автоматов, алгоритмизация, программирование, математическое моделирование) и прикладными знаниями (обработка текстов, создание электронных презентаций, обработка данных в электронных таблицах, коммуникация в цифровых средах, диссеминация информации в виртуальных пространствах и т. д.) становится очень подвижной.

Ответ на основной вопрос, с которым сталкивается любая общеобразовательная система, а именно, какие компетенции должна формировать и развивать школа: фундаментальные, стратегические, или прикладные, необходимые только на ближайший день? – наиболее актуален именно в случае *Информатики*, поскольку информационные технологии проникают практически во все сферы общественной жизни. Ситуация еще более осложняется тем фактом, что цифровые технологии меняются каждые 3-5 лет, а ориентация школьной программы по *Информатике* только на формирование и развитие компетенций по использованию современных ИКТ-инструментов приведет к тому, что указанные компетенции станут неактуальными за очень короткий период времени.

Куррикулум по *Информатике*, издание 2019 года, нацеливает рассматриваемую дисциплину на формирование именно фундаментальных цифровых компетенций, которые позволили бы учащимся и будущим выпускникам самостоятельно изучать и развивать навыки использования все новых и новых ИКТ-инструментов, которые появляются каждые 2-3 года.

В целом, структура и содержание Куррикулума по *Информатике*, издание 2019 года, разработаны в соответствии с подходами, используемыми в большинстве стран с традициями в области изучения данного предмета в системе общего образования. Акцент в этом нормативном документе сделан на формировании и развитии компетенций, основанных на знании фундаментальных достижений в области Информатики: алгоритмизация и программирование, математические основы Информатики, вычислительные методы, компьютерное моделирование и т. п. Вопросы, связанные с использованием цифровых инструментов (операционные системы, программные приложения для организации и обработки данных, компьютеры и сети и т. п.), рассматриваются концептуально, независимо от конкретных

технологий. Такой подход, по мнению большинства специалистов в области преподавания информатики, дает возможность самим учащимся развивать навыки самостоятельного усвоения новых ИКТ-инструментов.

В отличие от классических школьных дисциплин, в случае *Информатики* быстрые технологические изменения, социальные потребности и потребности бизнес-среды из промышленного сектора информационных и коммуникационных технологий требуют обеспечения большей гибкости Куррикулума. С этой целью, впервые в истории общего образования Республики Молдова, разработка Куррикулума по *Информатике*, основывалась на двух новых принципах, уже принятых педагогическим сообществом, но еще полностью не реализованных в предыдущих учебных программах: структурирование Куррикулума по модулям и расширение возможностей каждого из учебных заведений и даже каждого из учеников самостоятельно выбирать содержание и используемые для обучения ИКТ-средства.

1. Концептуальные основы Куррикулума по Информатике

1.1. Концепция Куррикулума по Информатике

Куррикулум по *Информатике* для лицейского образования служит основой для проектирования, организации и эффективного проведения образовательного процесса в контексте педагогики, ориентированной на формирование и развитие компетенций.

Как учителям, так и родителям важно знать суть основных понятий, используемых в указанном документе:

Ключевые/трансверсальные компетенции – ожидания общества от школьного образования и основные результаты, которые должны быть достигнуты каждым из учащихся в конце обучения. Кодекс об образовании устанавливает следующие ключевые компетенции:

- СС1. Компетенции общения на румынском языке.
- СС2. Компетенции общения на родном языке.
- СС3. Компетенции общения на иностранных языках.
- СС4. Компетенции в математике, науке и технике.
- СС5. Компетенции в цифровых технологиях.
- СС6. Компетенция научиться учиться.
- СС7. Социальные и гражданские компетенции.
- СС8. Предпринимательские компетенции и инициативность.
- СС9. Компетенции культурного самовыражения и осознания культурных ценностей.

Специфические компетенции дисциплины – интегрированная система знаний, навыков, отношений и ценностей, приобретенных, сформированных и развитых в процессе обучения, мобилизация которых позволяет идентифицировать и решать различные проблемы в разнообразных контекстах и ситуациях. Предполагается, что рассматриваемые компетенции должны быть достигнуты в конце лицейского образования.

Специфические компетенции школьной дисциплины Информатика вытекают из ключевых/трансверсальных компетенций и были сформулированы как в соответствии с *Основами Национального куррикулума*, так и в соответствии с рекомендациями основных европейских документов, касающихся построения информационного общества: *Информационные компетенции для XXI-го века (2007)*, *Цифровая повестка дня для Европы (2010)*, *План действий по цифровому образованию (2018)*.

Куррикулум по *Информатике* устанавливает следующие специфические компетенции:

- CS1. Использование цифровых инструментов для повышения эффективности процесса обучения и работы, демонстрируя инновационный и практический подходы.

- CS2. Взаимодействие с членами виртуальных сообществ в целях обучения и работы, проявляя интерес к активному обучению, исследованиям и сотрудничеству, соблюдая этику виртуальных сред.
- CS3. Распространение в цифровых средах результатов личных и командных достижений, доказывая изобретательность, командный дух и убежденность.
- CS4. Создание цифровых графических, видео и аудио продуктов, демонстрируя креативность и уважение по отношению к национальным и универсальным ценностям.
- CS5. Научное восприятие роли и воздействие информационных явлений на современное общество, проявляя критическое и позитивное мышление при интеграции различных областей познания и человеческой деятельности.
- CS6. Обработка данных экспериментов из области реальных и общественных наук, демонстрируя критическое мышление, ясность и корректность.
- CS7. Алгоритмизация методов анализа, синтеза и нахождения решений проблемных ситуаций, демонстрируя креативность и настойчивость.
- CS8. Внедрение алгоритмов с помощью сред разработки программ, демонстрируя сосредоточенность и настойчивость.
- CS9. Исследование проблемных ситуаций с помощью моделирования, планирования и проведения виртуальных экспериментов в цифровых средах, демонстрируя аналитический подход, ясность и лаконичность.

На протяжении всего процесса преподавания-учения-оценивания учитель должен обеспечивать взаимосвязь дидактического подхода к обучению и развитию специфических компетенций Информатики с развитием и консолидацией ключевых компетенций. Для этого учителю следует руководствоваться соотношениями между специфическими компетенциями школьной дисциплины *Информатика* и ключевыми компетенциями, изложенными в *Кодексе об образовании* (см. ниже-приведенную таблицу).

Таблица 1. Соотношения между специфическими компетенциями школьной дисциплины Информатика и ключевыми компетенциями

	CC1	CC2	CC3	CC4	CC5	CC6	CC7	CC8	CC9
CS1	+	+	+		×		+	+	+
CS2	+	+	+		×		+		+
CS3	+	+	+		×		+	+	+
CS4	+	+	+		×	+	+	+	+
CS5				+	×	+			
CS6				+	×	+		+	
CS7				+	×	+		+	
CS8				+	×	+		+	
CS9				+	×	+		+	

Легенда:

- × – специфическая компетенция непосредственно способствует формированию ключевой компетенции.
- + – специфическая компетенция косвенно способствует формированию ключевой компетенции.

В целом, Куррикулум по *Информатике* был разработан таким образом, чтобы на каждом занятии учащийся имел возможность, с одной стороны, формировать свои компетенции, предусмотренные *Кодексом об образовании*, а с другой – максимально раскрывать свой интеллектуальный потенциал и творческий подход в свете интерактивной дидактической стратегии, которая должна стимулировать любопытство учащихся и вовлекать их в мотивационный процесс обучения. Чтобы полностью использовать преимущества указанного подхода, важно, чтобы в процессе долгосрочного и краткосрочного дидактического проектирования преподаватель выбирал такие материалы для преподавания и такие учебно-оценочные мероприятия, которые обеспечивают не только формирование и развитие специфических компетенций в области компьютерных наук, но, через них, и развитие ключевых компетенций. При этом, преподавателям следует уделять особое внимание формированию транс- и междисциплинарных связей, руководствуясь для достижения этой цели рекомендации STEAM¹.

Концепция STEAM основана на обучении и развитии компетенций учащихся посредством дидактической деятельности с ярко выраженным прикладным характером, причем знания приобретаются одновременно с навыками их применения. Согласно концепции STEAM, обучение должно основываться главным образом на исследованиях, на решении проблем и на разработке проектов, которые должны иметь явно выраженный междисциплинарный характер.

Рекомендуется, чтобы подходы к обучению, основанные на концепции STEAM, разрабатывались и внедрялись не только учителем Информатики, но и всеми педагогами, которые преподают в соответствующем классе. С этой целью должны быть сформированы группы из учителей, которые определяют общие темы для исследований, решения проблем и разработки проектов. Указанные темы в обязательном порядке должны охватывать несколько школьных предметов. Очевидно, что при междисциплинарном подходе к обучению, учителя, преподающие отдельные дисциплины, должны обеспечить для каждого из школьных классов синхронизацию соответствующих этапов учебной деятельности.

1.2. Инновационные направления Куррикулума по *Информатике*

Лицейский гуррикулум по *Информатике* характеризуется следующими инновационными элементами:

1. Использование новой таксономии компетенций, соответствующей последним достижениям педагогических наук.
2. Включение в специфические компетенции по информатике базовых знаний из области компьютерных наук в гармоничном сочетании с прикладными средствами информационно-коммуникационных технологий.
3. Повышение степени привлекательности Куррикулума путем его структурирования по модулям и расширения компонентов по выбору, что позволит ученикам выбирать для дальнейшего изучения именно те области Информатики, которые они считают более полезными и/или более интересными для них.

¹ STEAM – это аббревиатура от английских слов: Science, Technology, Engineering, Arts and Mathematics (Наука, Технология, Инженерия, Искусство и Математика).

4. Инвариантность Куррикулума по отношению к специфике используемых программных продуктов (операционные системы, офисные приложения, программы обработки графики, аудио и видео, среды разработки компьютерного программного обеспечения и т. д.).
5. Упрощение Куррикулума путем исключения тем, которые доступны для усвоения только для очень небольшого числа учеников: определенные техники программирования, сложные численные методы, методы детального проектирования баз данных. Предполагается, что указанные темы могут изучаться по выбору или в кружках по информатике.
6. Повышение привлекательности Куррикулума путём включения в программы для каждого из старших классов модулей по выбору: обработка графики, аудио и видео информации; цифровая фотография; обработка данных в гуманитарных исследованиях, анализ и синтез информации из баз данных, элементы веб-дизайна, визуальное программирование.
7. Переосмысление процессов изучения алгоритмов, методов алгоритмизации и программирования, при этом учащиеся и преподаватели имеют возможность самостоятельно выбирать языки программирования и используемые среды для разработки программ.
8. Обновление тем, относящихся к изучению принципов работы и структуры компьютеров и цифровых сетей, путем включения в учебные программы основополагающих принципов Информатики. Чтобы глубже усваивать указанные принципы, в учебную программу было включено углубленное изучение современного цифрового оборудования и современных технологий, в частности тех, которые относятся к Интернету, к распределенной обработке данных (в «облаках»), к электронным услугам.
9. Расширение тем, связанных с соблюдением нормативно-правовой базы, правил цифровой безопасности, эргономики и цифровой этики при использовании современных средств информации и коммуникационных технологий.
10. Внедрение концепции STEAM.

Безусловно, внедрение инновационных подходов, перечисленных выше, требует от учителей Информатики обновления парадигм разработки долгосрочных и краткосрочных учебных проектов, переосмысления форм планирования и реализации всего комплекса дидактических мероприятий.

2. Куррикулум по Информатике и дидактическое проектирование

2.1. Куррикулум по Информатике как основа для долгосрочного дидактического проектирования

В контексте Куррикулума для лицейского образования центральной концепцией дидактического проектирования по информатике является персонализированный дидактический проект.

Персонализированный дидактический проект реализует право учителя принимать решения относительно тех обучающих действий, которые он считает оптимальными для обеспечения качества образовательного процесса в классе. В то же время, благодаря персонализированному дидактическому проекту, учитель получает возможность самостоятельно подбирать для каждого ученика индивидуализированный дидактический подход, исходя из его особенностей и конкретных условий учебного заведения.

Дидактическое проектирование требует от учителя заблаговременного обдумывания планируемых действий и событий в классе, предварительной конфигурации преподавания, обучения и оценивания, разделяя их на два взаимосвязанных уровня: долгосрочный и поурочный.

Как известно, в общеобразовательном учреждении документами поэтапного дидактического проектирования, разрабатываемые преподавателем и утвержденными в установленном порядке, являются:

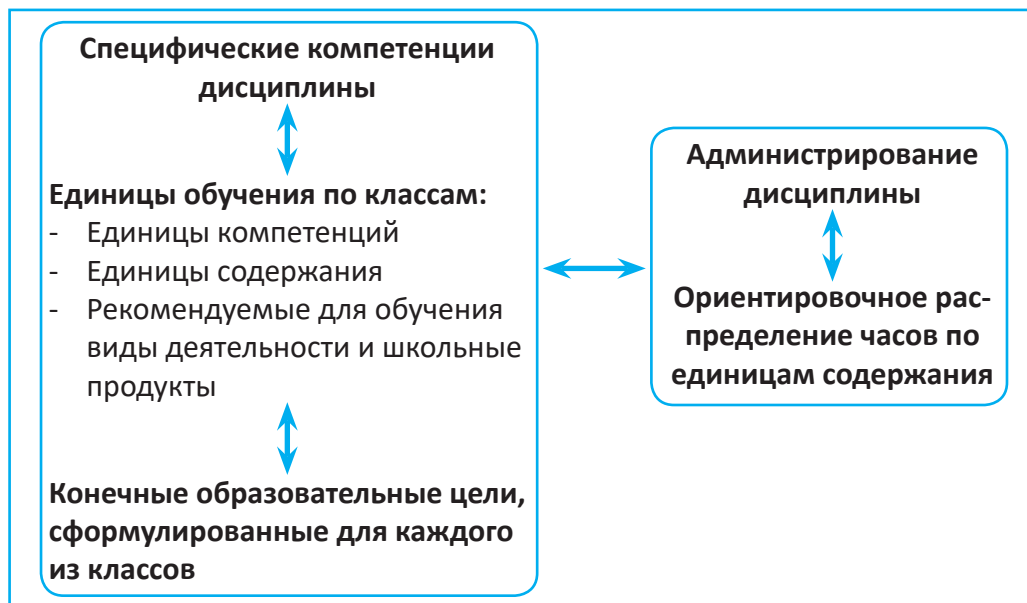
- *На уровне долгосрочного проектирования:* проект администрирования дисциплины (годовой, полугодовой); проекты единиц обучения;
- *На уровне краткосрочного проектирования:* проекты уроков или других организационных мероприятий, например, экскурсии на предприятиях с ИКТ-профилем.

Документы поэтапного дидактического проектирования являются административными документами. В них Куррикулум (школьная программа) внедряется индивидуализировано, с учетом конкретных условий учебного процесса в классе, в контексте конкретного распределения методологических, временных и материальных ресурсов, которые учитель считает оптимальными для соответствующего этапа.

Таким образом, Куррикулум по *Информатике* является основным нормативным документом для персонализированного проектирования дидактической деятельности в классе.

В этом смысле, учитель должен интерпретировать учебные программы каждого из классов с учетом взаимосвязи элементов, представленных на рисунке 1.

Рисунок 1. Взаимодействие элементов, направляющих дидактическое проектирование



Долгосрочное проектирование выполняется исходя из специфических компетенций, общих для всего лицейского курса по информатике.

Годовой дидактический проект предмета разрабатывается в соответствии с данными из раздела *Администрирование дисциплины* и с учетом ориентировочного распределения часов по единицам обучения.

Системы единиц компетенций, предусмотренные для каждой из единиц обучения, предназначены для возможного суммативного оценивания в конце соответствующей единицы обучения и выборочно – для текущего, формирующего оценивания. Именно системы единиц компетенций направляют дидактическое проектирование единиц обучения и краткосрочное дидактическое проектирование.

Единицы компетенций, приведенные в конце учебной программы каждого из классов, предназначены для возможного годового оценивания. Указанные единицы компетентностей являются основой для разработки инструментариев оценивания.

Единицы содержания являются информационными средствами, с помощью которых обеспечивается достижение систем компетенций, установленных для соответствующей единицы обучения. Одновременно, должны учитываться и необходимость достижения специфических компетенций дисциплины, а также трансверсальных/транс дисциплинарных компетенций.

Единицы содержания включают в себя списки специфических для Информатики терминов: слова/фразы, которые при усвоении соответствующей единицы обучения должны войти в активный словарный запас учащегося.

Виды деятельности, рекомендуемые для обучения и оценивания, представляют собой открытые списки важных для установления контекстов единиц компетенций, предназначенных для обучения/развития и оценивания в рамках соответствующей единицы обучения.

При разработке и проведения уроков преподавателю предоставлена свобода в использовании и даже в дополнении указанного списка, но одновременно на него возлагается и ответственность за его соответствие специфике единиц обучения, составу конкретного класса, имеющихся ресурсов и т. п.

2.2. Долгосрочное дидактическое проектирование

Посредством долгосрочного дидактического проектирования, ресурсы, учебные часы, виды учебной деятельности и т. п. распределяются в соответствии со спецификой компетенций, которые необходимо формировать и развивать у учащихся в рамках информатики. Долгосрочный дидактический проект обеспечивает целостное представление учебного процесса, в котором учитель и ученики будут вовлечены на протяжении длительного периода, позволяя тем самым разумно распределять временные ресурсы.

Прежде чем приступить к долгосрочному дидактическому проектированию, учитель, по согласованию с учениками и их родителями, должен определить – какой из модулей по выбору будет изучаться в текущем году. Для того чтобы правильно ориентировать учеников и родителей в выборе одного из модулей, предлагаемым в Куррикулуме по *Информатике*, учитель должен предварительно собрать, систематизировать и анализировать сведения, касающихся:

- предпочтения учеников и родителей, используя для этой цели опросники, желательно электронные;
- наличия в школе требуемых цифровых ресурсов (количество учеников за компьютером во время практических уроков по информатике, количество часов, в течение которых ученик может работать за компьютером вне уроков, скорость Интернет-соединений, доступность прикладных программ общего назначения, обеспеченность специализированными программными продуктами и т. п.);
- наличия традиционных и цифровых учебных материалов (учебников, образовательных программных продуктов, автоматизированных систем тестирования, веб-сервисов и т. п.);
- обеспечения доступа каждого из учеников к цифровым ресурсам, необходимых для изучения в полном объеме выбранного модуля.

Если в процессе выбора модуля, мнения учеников разделяются примерно в равных пропорциях, учитель может проводить одновременное обучение по двум модулям по выбору, используя при этом технологии обучения, основанные на индивидуальной и групповой учебной деятельности.

С целью оказания помощи учителям в разработке долгосрочных учебных проектов, в главе *Администрирование дисциплины* Куррикулума по *Информатике* содержится рекомендуемое распределение часов по модулям. Будучи гибким,

Куррикулум по *Информатике* предоставляет учителю определенную свободу в распределении учебных часов по модулям, однако уменьшение числа часов, выделенных для обязательных модулей, недопустимо.

После установления конкретного числа учебных часов, выделенных для каждого из модулей, учитель должен распределить соответствующие учебные часы по единицам обучения, руководствуясь при этом необходимостью обеспечения эффективности образовательного процесса.

При распределении учебных часов по учебным единицам, учитель должен учитывать следующие факторы:

- начальный уровень подготовки учеников к изучению соответствующего материала;
- степень сложности материала, планируемого для изучения в каждой из единиц обучения;
- степень сложности практических заданий, которые ученики должны выполнить на компьютере;
- долю текущего и итогового оценивания в предполагаемой обучающей деятельности.

После распределения учебных часов, учитель должен тщательно выбирать технологии обучения, используя для этого все разнообразие дидактических методов, подробно описанных в курсах общей и специальной дидактики и, в нашем случае, в курсе *Дидактика Информатики*.

В качестве примера, в Приложении 1 представлен долгосрочный дидактический проект для 10-го класса, разработанный на основе Куррикулума по *Информатике*, издание 2019 года.

В целом, долгосрочное дидактическое проектирование должно выполняться в рамках комплексного подхода, обеспечивающего взаимосвязь между единицами компетенций, тематическим содержанием, обучающими мероприятиями и школьными продуктами. Учебная нагрузка обучаемого должно быть равномерно распределена по семестрам и неделям учебного года в соответствии с Куррикулумом.

В процессе долгосрочного дидактического проектирования особое внимание следует уделять формированию и постепенному развитию специфических компетенций информатики, обеспечивая логическую преемственность единиц содержания.

2.3. Краткосрочное дидактическое проектирование

Краткосрочный дидактический проект представляет собой хорошо обдуманную и персонализированную схему урока, которая содержит:

- то, чего преподаватель стремимся достичь – цели урока;
- элементы, необходимые для достижения целей урока – содержание и стратегии обучения (формы, методы, процессы, средства обучения);
- инструменты для измерения эффективности обучения – стратегии оценивания.

В литературе по специальности приведены разнообразные модели проектов уроков, однако все они имеют примерно одинаковую структуру. Преподаватель

имеет право выбрать ту модель, которую он считает наиболее подходящей и эффективной для каждого из планируемых уроков.

Независимо от используемой модели, основываясь на опыте пилотного внедрения Куррикулума по *Информатике*, издание 2015 г., рекомендуется, чтобы проект урока включал следующие разделы:

Общие данные. Календарная дата, класс, дисциплина.

Тема урока. Тему урока следует взять из долгосрочного дидактического проекта.

Тип урока. Укажите соответствующий тип урока, исходя из необходимости формирования и развития компетенций:

- приобретение знаний;
- понимание знаний;
- применение знаний;
- анализ и обобщение знаний;
- оценка знаний;
- смешанный.

Единицы компетенций. В соответствии с Куррикулумом устанавливаются приоритетные для данного урока единицы компетенций, которые планируется формировать или развивать.

Цели урока. В зависимости от конкретной ситуации, исходя из единиц компетенций, установленных для данного урока, формулируются 4-6 целей урока. Цели урока должны покрывать следующие области:

- когнитивная область (усвоение знаний, формирование интеллектуальных способностей);
- аффективная область (формирование убеждений, чувств, отношений);
- психомоторная область (выполнение операций по обработке информации, тренировка моторики, формирование практических навыков).

Выбор глаголов действия выполняется в соответствии с рекомендациями Куррикулума по *Информатике*: таксономий Блума-Андерсона (*Bloom-Anderson*) и Блума-Кратвола (*Bloom-Krathwohl*) для когнитивной области; таксономии Кратвола (*Krathwohl*) для аффективной области; таксономий Дейва (*Dave*) и Симпсона (*Simpson*) для психомоторной области. Соответствующие таксономии можно найти в литературе по специальности.

При формулировании целей урока следует учитывать необходимость формирования и развития установок и ценностей, характерных для информатики, которые явным образом указаны в специфических компетенциях дисциплины.

Стратегии обучения. Этот раздел должен содержать:

- организационные формы (фронтальная, индивидуальная, групповая);
- методы, процедуры и методики обучения (как классические, так и современные, основанные на широкомасштабном использовании компьютерных средств обучения);
- учебные ресурсы (демонстрационные, индивидуальные, раздаваемые ученикам)².

² STEAM – это аббревиатура от английских слов: *Science, Technology, Engineering, Arts and Mathematics* (Наука, Технология, Инженерия, Искусство и Математика).

Стратегии оценивания. Этот раздел должен включать:

- тип оценивания;
- инструменты для оценивания, оцениваемые продукты, критерии оценивания;
- процедуры самооценивания/взаимного оценивания.

При разработке этого раздела следует учитывать, что стратегии оценивания могут быть:

- инструментальными – оценивание осуществляется в специально созданных условиях, которые включают разработку и применение инструментов оценивания, построенных на основе продуктов (тестирование, сопровождаемое матрицей спецификаций и шкалой проверки/верификации/оценивания; устное, письменное, практическое или комбинированное тестирование и т.п.);
- неинструментальными – оценивание осуществляется в обычных условиях на основе наблюдений за учебной деятельностью учащихся и немедленным обратным получением данных, без использования инструментов оценивания. Указанные стратегии предназначены для повышения осведомленности партнеров, участвующих в процессе обучения (преподавателей и учеников) об учебных достижениях учащихся, о прогрессе в формировании компетенций, о трудностях, с которыми сталкивается каждый из учащихся, о путях преодоления выявленных трудностей и повышении эффективности учебного процесса.

Библиография. В этом разделе следует включать только те источники, которые доступны для учащихся именно данного учебного заведения. Особое внимание следует уделять онлайн-ресурсам, которые должны соответствовать правилам цифровой этики и требованиям цифровой безопасности.

Дидактический сценарий. Проект урока может быть реализован как на основе пошаговых указаний, выбранных в соответствии с типом урока, так и на основе этапов из модели урока *Вызов – Осмысление – Рефлексия – Расширение*.

Степень детализации сценария урока устанавливается по согласованию с руководством учебного заведения исходя из квалификации преподавателя (наличие или отсутствие дидактической степени) и результатов его предыдущей профессиональной деятельности.

В зависимости от выбранной модели проекта и его степени детализации, возможно использование различных вариантов организации дидактического сценария: текстового или табличного. Как текстовая, так и табличная версии сценариев должны содержать явную информацию о действиях учителя и учеников на каждом из этапов урока, ожидаемые от учеников школьные продукты и критерии их оценивания.

В качестве примера, в Приложении 2 представлен краткосрочный дидактический проект для 10-го класса, разработанный на основе Куррикулума по *Информатике*, издание 2019 года.

3. Куррикулум по Информатике и методологические основы учебного процесса

3.1. Логика и принципы разработки дидактических стратегий

Типология и специфика дидактических стратегий. Стратегия обучения – это термин, объединяющий задачи и учебные ситуации, представляя собой целостную систему образовательных ресурсов, методов и процедур, направленных на формирование и развитие определенных компетенций. Дидактическая стратегия занимает центральное место в учебной деятельности, потому что проектирование и организация урока осуществляются в соответствии со стратегическими решениями учителя. Она задумана как сложный дидактический сценарий, в котором участвуют субъекты преподавания/обучения, условия осуществления и соответствующие методы. Таким образом, стратегия устанавливает оптимальный методический маршрут для решения конкретной ситуации преподавания/обучения. В этом смысле, стратегическое проектирование может предотвратить ошибки, риски и нежелательные события в учебной деятельности.

В качестве составляющих элементов, методы преподавания/обучения должны соответствовать выбранной стратегии. Однако, стратегию не следует путать с методом или дидактической методологией, поскольку последние направлены только на деятельность по обучению – учению – оцениванию, в то время как дидактическая стратегия нацелена на процесс обучения в целом.

Основными компонентами стратегии обучения являются:

- система форм организации и ведения образовательной деятельности;
- методологическая система (методы и процедуры обучения);
- система образовательных ресурсов (используемые ресурсы);
- система операционных целей/единиц компетенций (для урока/для единицы обучения).

Среди наиболее важных стратегий преподавания, применяемых в процессе изучения Информатики, отметим:

- индуктивные стратегии (дидактический подход от частного к общему);
- дедуктивные стратегии (дидактический подход от общего к частному);
- аналоговые стратегии (обучение на основе моделей);
- смешанные стратегии (индуктивно-дедуктивные и дедуктивно-индуктивные);
- алгоритмические стратегии (демонстрационные, интуитивно понятные, структурированные, упорядоченные);
- эвристические стратегии (получение знаний своими силами – проблематизация, экспериментирование, анализ, синтез).

Наиболее часто специалисты в области преподавания Информатики используют алгоритмические, смешанные и аналоговые стратегии, комбинируя их с элементами самоуправления и самостоятельности, с упором на полу-управляемое

преподавание-обучение. Для того чтобы ученики достигли успехов в области Информатики, рекомендуется использование эвристических стратегий обучения, с их ориентацией на управляемое самообучение.

Дидактические стратегии реализуются с помощью информативных и активно-партисипативных методов обучения и учения, методов самостоятельного изучения, проверки и самооценивания.

Принципы разработки дидактических стратегий. Технологические достижения Информатики ведут к сложным изменениям всех процессов, характерных для человеческой деятельности. В частности, они влияют и на сферу образования. Модели и процессы трансформируются, появляются новые и новые знания, предыдущие знания становятся менее актуальными. Снижается возрастной предел, с которого начинается изучение и использование цифровых технологий.

В результате модернизации структуры учебных процессов, Куррикулума, моделей и форм обучения, следует модернизировать и алгоритмы разработки дидактических стратегий, особенно в случае школьной Информатики. В частности, дидактические стратегии по Информатике должны учитывать специфику модульного обучения, обеспечивать интеграцию Информатики с другими дисциплинами из области точных и гуманитарных наук (STEM / STEAM).

Поэтому стратегии обучения следует разрабатывать в соответствии со следующими принципами:

Актуальность. Стратегии обучения должны быть основаны на последних достижениях в области Информатики и смежных наук, сосредоточены на новейших инструментах и информационных технологиях, адаптируя при этом элементы новизны к конкретным возрастным особенностям учащихся.

Модульность. Стратегии обучения должны быть сгруппированы в соответствии со структурой учебных модулей, определяя деятельность по обучению и оцениванию в соответствии с конкретным содержанием каждого из модулей. В частности, для модулей, ориентированных на формирование навыков по разработке компьютерных программ или по программированию цифровых устройств, главным образом должны использоваться алгоритмические и эвристические стратегии. При изучении «технологических» модулей, следует использовать аналоговые стратегии. Индуктивные и дедуктивные стратегии должны использоваться при изучении всех модулей по выбору.

Междисциплинарность/трансдисциплинарность. Для стратегий, предназначенных для усвоения определенных тем, будут заранее определены их связи с дисциплинами STEM/STEAM, а цели конкретных обучающих действий должны быть согласованы с куррикулярными требованиями для соответствующих школьных дисциплин. Примеры: использование бинарного поиска для решения алгебраических уравнений; программирование роботизированных устройств для выполнения повторяющихся действий.

Гибкость. Разработанные стратегии должны быть адаптированы к специфическим условиям учебного процесса и индивидуальным особенностям обучаемых учеников, таким как:

- психолого-педагогический контекст;
- личностный контекст, как учеников, так и учителей;

- цифровая инфраструктура учебного заведения;
- стиль преподавания и т.п.

3.2. Дидактические стратегии формирования специфических компетенций по Информатике

Методические рекомендации и способы разработки стратегий обучения. Основными компонентами, определяющими дидактическую стратегию являются:

- формы образовательной деятельности;
- методы обучения, используемые в образовательной деятельности;
- средства, используемые учителем/учениками во время занятий;
- цели, преследуемые учителем/учениками.

Следовательно, для разработки стратегии обучения необходимо выбрать соответствующие формы, методы, средства и цели, а также обеспечить их взаимную согласованность на всем протяжении реализации разрабатываемой стратегии.

Формы дидактической деятельности. Формы дидактической деятельности классифицируются по различным критериям, основными из которых являются:

- продолжительность деятельности: учебный год, семестр, учебная единица, урок, момент урока;
- целевая группа: класс, группа из нескольких учеников, отдельный ученик;
- способ организации:
 - а) формальный (с систематическим, однородным, запрограммированным, дозированным информационным потоком информации, сопровождаемый непрерывным и систематическим формирующим оцениванием для достижения образовательных целей и психического развития учеников);
 - б) неформальный (школьные и внеклассные мероприятия, в которых образовательные мероприятия структурированы, организованы и институционализированы вне традиционной системы обучения, будучи разнообразными, гибкими, создавая возможности для удовлетворения конкретных интересов, развития индивидуальных способностей каждого из учащихся);
 - с) информальный (поток информации является случайным, неорганизованным, ненаправленным, несистематическим; полученные знания или навыки являются случайными, ненамеренно приобретенными в результате повседневной деятельности).

Таким образом, форма дидактической деятельности устанавливает временную, пространственную и организационную структуру, в которой интегрированы методы, используемые для обучения, и средства обучения, используемые во время дидактической деятельности.

Методы обучения. Методы обучения – это способы, с помощью которых учащиеся, направляемые педагогами, приобретают знания и навыки, развивают общие интеллектуальные и специфические способности.

Метод представляет собой совокупность умственных и практических операций, посредством которых учащийся, с помощью учителя или самостоятельно,

раскрывает суть событий, процессов и явлений. В широком смысле, метод является рационализированной практикой, обобщением, подтвержденным текущим опытом или психолого-педагогическим экспериментом, который служит для преобразования и улучшения человеческой природы.

Основными функциями методов обучения являются:

- когнитивная – управление знаниями с целью получения новых знаний;
- нормативная – способ обучения-учения;
- мотивационная – стимулирование интереса к обучению;
- формативно-воспитательно-компенсаторная – упражнения, тренировки и развитие психических процессов.

Основными методами обучения являются дидактическое изложение и беседа, демонстрация, наблюдение, работа с учебником, упражнение, алгоритмизация, дидактическое моделирование, проблемное обучение, программированное обучение, тематическое исследование, имитация, обучение через открытие, исследовательский проект. Большинство методов хорошо известны и описаны в литературе по специальности. Ниже мы остановимся на описании только тех методов, которые рекомендуется применять более часто при внедрении в учебном процессе, ориентированном на компетенции Куррикулума по *Информатике*.

Проблемное обучение. Этот метод называется также обучением путем решения задач. Рабочее определение рассматриваемого метода: дидактический метод, заключающийся в том, что он ставит перед учеником некоторые сознательно созданные проблемные ситуации, разрешая которые, за счет собственных усилий, ученик узнает что-то новое.

В соответствии с литературой по специальности, проблемными ситуациями, которые могут быть предложены ученикам на уроках Информатики, являются те, в которых:

- существует несоответствие между предыдущими знаниями ученика и теми, которые ему необходимы для решения новой ситуации;
- ученик должен выбирать из цепочки или системы знаний, даже неполных, только те, которые необходимы для решения конкретной ситуации, пополняя тем самым недостающие знания;
- ученик сталкивается с противоречием между теоретически возможным способом решения и трудностью его применения на практике;
- ученик должен выявить динамику движения в системе изначально кажущейся статичной;
- ученик должен применять в новых условиях ранее усвоенные знания.

Необходимость применения указанного метода легко доказать, учитывая его многочисленные преимущества, которые подходят для современного учебного процесса: метод благоприятствует формирующему аспекту образования путем эффективного и поддерживаемого участия ученика, развития его интереса к познанию; метод повышает применимость информации, усвоенной учеником для решения практических задач, с которыми он сталкивается в реальной жизни; метод создает для ученика отличную возможность применения усвоенных знаний для решения новых задач.

Пример внедрения проблемного обучения представлен в Приложении 3.

Проектное обучение. Проектное обучение – это метод обучения и оценивания, который позволяет ученикам приобретать основные знания и практические навыки, необходимые для успешной карьеры. Если мы действительно сосредоточены на достижении основных образовательных целей XXI века, то этот метод должен быть в центре внимания.

Тщательное и глубокое обучение методом проекта организуется на основе открытого вопроса или ситуации. Таким образом, ученики сосредотачиваются на углубленном изучении важных тем, на дебатах, вопросах и/или на решении учебных проблем.

Метод мобилизует учащихся на усвоение теоретических знаний и на приобретение специфических компетенций. Типичное обучение на основе метода проектов (и большинство занятий) начинается с представления ученикам знаний и концепций, а по мере их усвоения, создаются возможности для их практического применения. Реализация проекта начинается с представления конечной цели, для достижения которой ученикам требуются новые, конкретные знания и концепции, что создает предпосылки для обучения.

Основными преимуществами проектного обучения являются:

1. Учащиеся решают не абстрактные, а реальные задачи из окружающего их мира; по возможности, они сами выбирают или определяют существенные для них проблемы.
2. Занимаясь учебно-исследовательской деятельностью, учащиеся развивают навыки планирования, критическое мышление, навыки идентификации и решения проблем.
3. Специфические знания лучше усваиваются, а специфические умения лучше формируются именно в контексте выполнения конкретных работ из проектных заданий.
4. Метод облегчает формирование и развитие навыков межличностного общения, совместной работы в коллективе.
5. Метод позволяет ученикам усовершенствовать на практике навыки, необходимые для профессионального и карьерного роста (планирование времени и ресурсов, воспитание чувства личной ответственности и т. п.).
6. Метод напрямую ориентирован на достижение куррикулярных целей обучения, явно указанных в постановке задачи на выполнении проекта.
7. Метод включает и деятельность учеников по рефлексии относительно критического анализа, полученного в рамках проекта опыта и соотнесения полученного опыта с конкретными стандартами обучения.
8. Проекты завершаются презентациями или школьными продуктами, подтверждающими выполнение учебных задач.

Пример внедрения проектного обучения представлен в Приложении 4.

Обучение на основе тематического исследования. Этот метод состоит в столкновении ученика с реальной жизненной ситуацией, посредством которой наблюдение, сбор данных, их интерпретация, выявление и понимание причинно-следственных связей ведут к приобретению знаний. Метод включает следующие этапы:

1. Выбор темы исследования и выделение ее основных, значимых элементов.
2. Формулировка целей исследования, которые должны быть установлены исходя из проблемной ситуации.
3. Сбор информации в связи с исследуемой темой.
4. Систематизирование и анализ собранного материала, прибегая при этом к разнообразным методам, в том числе и статистическим.
5. Анализ и обсуждение собранных данных, которые могут происходить в различных организационных форматах.
6. Формулировка выводов и их осмысление: анализ ситуации и проверяемых гипотез, применяемых в исследовании методов, значимость полученных выводов.

В целом, тематическое исследование представляет собой составной метод, включающий в себя целый набор других методов, без которых исследование не может быть выполненным. Для тематических исследований в Информатике, особенно в случае когда ученики выполняют их в первый раз, желательно чтобы этапы 1 и 2 выполнялись учителем, а следующие – учениками, но при существенной помощи учителя.

Пример обучения на основе тематического исследования приведен в Приложении 5.

Средства обучения. Средства обучения представляют собой набор ресурсов (предметы, инструменты, продукты, устройства, оборудование и технические системы), которые поддерживают и облегчают учебные и оценочные мероприятия в процессе преподавания и учения.

Классификация образовательных ресурсов производится в соответствии с запрашиваемым анализатором, а также характером соответствующих ресурсов. Таким образом, ресурсы можно классифицировать на:

- визуальные ресурсы: тексты, презентации, изображения и устройства их отображения;
- аудио-ресурсы: звукозаписи и устройства для их создания и воспроизведения;
- аудиовизуальные ресурсы: видеозаписи со звуковым сопровождением и устройства для их создания и воспроизведения.

Для учебной деятельности по дисциплине *Информатика* материальным воплощением учебных ресурсов являются цифровые устройства для обработки информации и специализированное периферийное оборудование, но не исключается и использование традиционных средств обучения, таких как карточки, таблички, физические модели, регистраторы, видеокамеры и т. д.

Правильное, сбалансированное использование средств обучения, позволяет обеспечить соблюдение следующих требований, предъявляемых к учебному процессу:

Гибкость – возможность выбора из множества доступных ресурсов и устройств тех, которые являются наиболее подходящими для конкретной на данный момент учебной деятельности.

Универсальность – возможность кодировать передаваемые сообщения в разных формах, с тем, чтобы одновременно задействовать несколько систем приема информации.

Параллельность – возможность одновременного использования одного и того же ресурса для нескольких учебных целей или несколькими пользователями.

Специфика дисциплины *Информатика* требует массового использования компьютера в качестве средства обучения. Являясь универсальным инструментом, компьютер, в зависимости от подключенных к нему периферийных устройств, от набора доступных электронных образовательных ресурсов и программных приложений, может заменить все традиционные технические средства обучения. Доступ к информационным ресурсам Интернета существенным образом расширяет возможности использования компьютера в образовательных целях.

При этом необходимо учесть, что среда обучения, пусть даже цифровая, сама по себе не обучает, а является лишь одним из многочисленных компонентов сложной системы, состоящей из форм, методов, средств и ресурсов, предназначенных для обучения, которые дополняют друг друга и которые должны быть интегрированы преподавателем в эффективную стратегию обучения и самообучения.

3.3. Стратегии и инструменты для оценивания результатов обучения

Оценивание компетенций. Процесс оценивания подлежит постоянной модернизации. Причинами для этого являются: непрерывная интеграция цифровых технологий в образовательные процессы, разработка и внедрение новых образовательных стратегий, внесение изменений в национальные и международные нормативные акты. Поскольку новый Куррикулум внедряет методы оценивания, основанные на анализе школьных продуктов, оценивание теряет свой особый статус и становится неотъемлемой частью процесса обучения. Элементы оценивания присутствуют во всех дидактических мероприятиях, которые проводятся в классе, а иногда, благодаря новым образовательным моделям, и за его пределами. Таким образом, оцениванию присваивается новая роль в выработке на основе точных, заранее установленных критериев, оценочные суждения о том, что и как ученик усвоил. Поэтому модели оценивания должны быть адаптированы к стратегиям обучения, используемым учителем.

Модели оценивания, предлагаемые в Куррикулуме, используют «критериальный» принцип оценки, то есть они предоставляют функциональную информацию; мобилизуют учеников для достижения общих целей обучения и для создания школьных продуктов; предлагают на каждом этапе обучения указания по улучшению обратной связи. Используемые в Куррикулуме модели оценивания имеют «корректирующий» характер, что позволяет ученику улучшать на основе полученных отзывов создаваемый им школьный продукт.

Направленность образовательного процесса на формирование школьных компетенций потребовало внедрения нового принципа оценивания: «осознанное» или «формирующее» оценивание, основанное на когнитивной психологии и на интеграции оценивания в учебный процесс. В соответствии с этим принципом, оценивание находится между *обучением* как набор процессов, посредством которых ученик приобретает и применяет знания, и *обучением учиться* как процесс

приобретения «знаний о самопознании». Формирующее оценивание соответствует доминирующему в настоящее время педагогическому подходу, основанному на активном участии и автономии ученика, на предоставлении ему четких ориентиров для улучшения собственной учебы исходя из его собственных трудностей и возможных пробелов в усвоенном им материале.

В заключение, современное оценивание:

- имеет формирующий характер и основано на четко определенных критериях (стандартах);
- органично вписывается в деятельность по обучению и разработке школьных продуктов, адаптируя процесс обучения к индивидуальным особенностям каждого из учеников.

Типы оценивания. Существует несколько критериев для определения типологии оценивания. Наиболее часто используемыми критериями являются цели и частота оценивания. В зависимости от указанных критериев, различаются следующие типы оценивания:

Первичное оценивание, которое устанавливает стартовый уровень подготовки учащегося перед определенным этапом обучения: в начале учебного года, семестра, учебной единицы и т. п., а также данные, которые необходимы для адаптации стратегии обучения к начальному уровню подготовки учащихся. Для первичного оценивания рекомендуется использовать компьютерное тестирование с помощью локальных или онлайн-тестов, без выставления оценок.

Непрерывное (формирующее, формативное) оценивание, которое состоит в текущем контроле и оценивании знаний, умений и навыков. Переход к следующему этапу обучения осуществляется только после получения положительных результатов текущего оценивания. При необходимости, на основе результатов непрерывного оценивания, принимаются меры по улучшению процесса обучения и оказания дополнительной помощи определенным ученикам. Дополнительная помощь может быть оказана индивидуально или в составе малых групп учеников. Непрерывное оценивание выполняется с помощью устных интервью, решения упражнений и задач на компьютере, наблюдением за процессом выполнения учениками каждого из ключевых этапов учебных проектов.

Суммативное (итоговое) оценивание осуществляется на различных учебных этапах (по окончании модуля, семестра, учебного года и т. д.) с помощью всеобъемлющих заданий, охватывающих всю рассматриваемую тематическую область. В конце учебного модуля можно использовать онлайн или локальное тестирование. Для организации семестровых работ следует руководствоваться методикой, приведенной в нормативных документах министерства. Организация общенационального тестирования (бакалавриат по информатике) осуществляется в соответствии с программой экзамена по дисциплине, а форма организации тестирования устанавливается нормативными документами министерства. Безусловно, подготовка учеников к бакалавриату должно осуществляться в соответствии с программой экзамена по дисциплине *Информатика*, утвержденной министерством. Естественно, указанная программа не включает тематику модулей по выбору, методика суммарного оценивания остается на усмотрение каждого преподавателя. В общем случае, при определении модуля по выбору, методика суммарного

оценивания должна основываться на школьных продуктах, созданных учениками: презентации, текстовые документы, компьютерные модели в виде электронных таблиц, компьютерные программы, цифровые мультимедийные представления, Веб-страницы, отчеты о проведенных исследованиях, разработанные проекты и т. п.

Динамика результатов оценивания, полученных за определенный период времени (периода обучения), служит основой для определения прогресса учащихся в формировании и развитии специфических компетенций, предусмотренных для рассматриваемого периода в соответствующих календарных планах.

Методы оценивания. Метод оценивания – это способ, при помощи которого учитель дает учащимся возможность продемонстрировать уровень приобретенных ими знаний и навыков. Уровень специфических компетенций, приобретённых учениками оценивается с помощью различных инструментов, которые необходимо выбирать исходя из цели оценивания. В случае школьной дисциплины «Информатика», рекомендуемые методы оценивания включают:

Традиционные методы оценивания, используемые в течение длительного времени в школьном образовании. Наиболее распространенными из них являются: устная проверка, письменная проверка, тесты, практические работы. Как правило, в случае Информатики, оценивание с помощью тестов, а также путем выполнения практических работ, должно осуществляться на компьютерах.

Устная проверка проводится на основе беседы, в течение которой учитель определяет усвоенные учениками знания. Беседа может быть индивидуальной, фронтальной или комбинированной. Основным преимуществом устной проверки является получение немедленной обратной связи. Метод также способствует развитию у учеников навыков выражения. Недостатком метода является зависимость результатов проверки от множества случайных факторов, которые могут влиять на ответы ученика: эмоциональное состояние учителя/ученика, неодинаковая степень сложности вопросов, психическое состояние или особенности характера оцениваемых. В случае школьной дисциплины Информатика, указанный метод рекомендуется использовать только в качестве вспомогательного инструмента с целью корректировки текущей учебной деятельности в соответствии с результатами обучаемых. Например, вопрос о количестве информации, а также последующее обсуждение с учениками единицы измерения количества информации, помогут им избежать блокирующие ситуации при решении упражнений и задач, в которых требуется вычисление количества информации в текстовых документах.

Письменное оценивание состоит в выполнении контрольных работ, мини-эссе, тезисов, тестов национальных экзаменов. В рамках рассматриваемого типа оценивания, прямой контакт ученика с проверяющим исчезает, а в некоторых случаях оценивание является даже внешним, что позволяет уменьшить влияние большинства субъективных факторов. Преимущества метода состоят в возможности проверки относительно за короткий промежуток времени большого числа учеников, проверки работ и выставления баллов в соответствии с заранее установленными критериями (шкала, схема начисления баллов), создание благоприятных условий для учеников, имеющих трудности при устном выражении и т.п. В случае школьной дисциплины *Информатика*, письменное оценивание с помощью бумажных

носителей допускается только в случаях отсутствия достаточного числа компьютеров.

Выполнение практических заданий/работ. Рассматриваемый метод оценивания направлен на проверку функциональности полученных учеником знаний, полноты их усвоения, интериоризации и экстериоризации. В случае школьной дисциплины *Информатика* указанный метод состоит в решении учеником специально подобранных или составленных преподавателем задач с последующим внедрением полученных решений в цифровых средах. При текущей учебной деятельности, оценивание выполняется путем проверки предлагаемых учеником решений, а на заключительном этапе – путем анализа объяснений и интерпретаций результатов, представленных учеником.

Очевидно, при применении традиционных методов оценивания необходимо обеспечить баланс между устными, письменными и практическими методами оценивания.

Методы формирующего оценивания обеспечивают индивидуализацию обучения, ориентированы на ученика и на создание оптимальных условий для его развития. Рассматриваемые методы следует интегрировать в среднесрочный (1-2 недели) или долгосрочный (до одного семестра) учебный процесс и использовать для получения информации, предназначенной для промежуточного или конечного оценивания результатов, достигнутых учеником или группой учеников. В последние годы указанные методы перешли из категории дополнительных в категорию основных методов оценивания и стали важной частью всего процесса оценивания. В случае школьной дисциплины *Информатика*, наиболее рекомендуемыми методами формирующего оценивания является реферат, тематическое исследование, проекты, портфолио, самооценивание.

Формирующие методы подвержены повышенному риску «повторного использования» учениками продуктов, ранее созданных другими лицами, что рождает такое явление как плагиат. Чтобы предотвратить появление плагиата, учителю следует осуществлять целый ряд мероприятий, от дискуссий по цифровой этике и авторскому праву до использования передовых методов и средств для поиска информации, идентификации источников, откуда «заимствованы» материалы, представленные учениками, применяя для этого современные компьютерные программы, предназначенные для автоматизации процесса обнаружения плагиата.

Критерии оценки для методов формирующего оценивания разрабатываются на основе показателей эффективности, приведенных в стандартах оценивания для дисциплины *Информатика*, используя при этом соответствующие таксономии (Блума-Андерсона, Симпсона, Дейва, Кратвола) и учитывая степень сложности ожидаемых школьных продуктов и длительность процесса их создания. В частности, рефераты и тематические исследования могут оцениваться по их завершению, в то время как проекты и портфолио учеников могут оцениваться и по мере их реализации, по завершению определенных, заранее выбранных этапов.

Техники оценивания. Техника оценивания представляет собой способ, с помощью которого учитель ориентирует учащихся на выработку ответов на заданные темы/вопросы. В зависимости от типов итемов, которые он включает, для тестов

со многими итемами/вопросами может потребоваться несколько техник оценивания.

Для традиционного тестирования (устное, письменное, компьютерное) следует использовать разнообразные типы итемов. Ниже представлена краткая характеристика основных типов итемов и рекомендации по их использованию.

Категория *объективных типов* итемов включает: итемы с множественным выбором, итемы с двоичным выбором и итемы на установление соответствия.

Итем с множественным выбором состоит из вопроса/запроса и множества ответов, из которых только один является правильным. Именно этот ответ и должен быть выбран учеником исходя из сущности вопроса/запроса указанного в итеме. Итемы рассматриваемого типа используются для измерения результатов обучения низшего таксономического уровня: запоминание определений, терминов, фактов, принципов, методов и процедур. На среднем и более высоком таксономическом уровнях (воображение, интериоризация), итемы с множественным выбором можно использовать для оценки способностей выявлять факты и принципы, интерпретировать причинно-следственные связи, аргументировать методы и процедуры.

Пример.

Выберите написание десятичного числа 124 в восьмеричной системе счисления:

- a) 174; b) 172; c) 164; d) F6.

Итемы с двоичным выбором требуют от ученика выбор одного из двух возможных ответов: истина/ложь; верно/не верно; да/нет; согласен/не согласен и т.п. Они могут использоваться на низшем таксономическом уровне для распознавания определенных терминов, конкретных данных, принципов, различий между утверждениями, основанными на фактах или мнениях.

Пример.

Выберите значение истинности утверждения “Десятичная система счисления (на основе арабских цифр) является НЕПОЗИЦИОННОЙ”:

- a) ИСТИНА б) ЛОЖЬ

Итемы на установление соответствия требуют установление соответствий/ассоциаций между словами, предложениями, фразами, буквами или другими категориями символов, расположенными в двух столбцах. Элементы из первого столбца называются посылками, а элементы из второго столбца представляют ответы. Критерий или критерии, на основании которых определяется правильные ассоциации, изложены/объяснены в указаниях, предшествующих этим двум столбцам. Как правило, итемы рассматриваемого типа используются для измерения уровня знаний о фактах, способности идентифицировать отношения между двумя предметами/понятиями/символами и т.п. Итемы на установление соответствия могут содержать различные типы отношений: термины/определения;

правила/примеры; символы/концепции; принципы/классификации и т. п. Очевидно, в качестве посылок и ответов можно также использовать рисунки или другие графические представления.

Пример.

Установите с помощью стрелок соотношение между понятиями из левого столбца с определениями в правом столбце:

Понятия	Определения
Абсолютная погрешность	Отношение абсолютной погрешности к модулю точного числа
Относительная погрешность	Модуль разницы между точным и приближенным значениями

Категория *полуобъективных итемов* включает итемы с коротким ответом, итемы с запросом дополнения, структурированные вопросы.

Итемы с коротким ответом разделяются на следующие подкатегории:

- С классическим вопросом, подразумевавшим получение единственного или множественного ожидаемого ответа.
- С упражнением, процесс выполнения которого генерирует ровно один результат.
- С индуцированным текстом. Из-за своей специфической структуры в информатике итемы рассматриваемого типа используется очень редко.

Пример итема с коротким множественным ответом.

Какую технику программирования можно применить для точного решения задачи о рюкзаке?

Очевидно, возможными ответами являются: Полный перебор, Перебор с возвратом.

Итемы с запросом на дополнение представляют собой утверждения, в которых пропущено одно или несколько значимых слов. Ученику предлагается найти и вписать эти слова в указанные места.

Пример.

Впишите пропущенные слова:

При переводе целого числа из двоичной в восьмеричную систему счисления число разбивается на группы из _____ цифр, начиная с _____.

Структурированные вопросы состоят из нескольких под-вопросов объективного типа, полуобъективного типа или мини-эссе, связанных между собой общим элементом. Они занимают нишу между техниками оценивания со свободными (открытыми) ответами, требуемые в итемах субъективного типа, и техниками с ограниченными (закрытыми) ответами, требуемые в итемах объективного типа.

Структурированные вопросы могут быть использованы для оценки знаний о принципах работы компьютера и компьютерных сетей, о способах организации информации на носителях для устройств внешней памяти, о способах общения человека с компьютером. Подвопросы могут охватывать практически все таксономические категории, начиная от простого воспроизведения (определения, перечисления, понятия и т. д.) до применения знаний, анализа, синтеза, формулирования гипотез и оценочных суждений.

Пример структурированного вопроса:

В чем суть алгоритма генерации всех подмножеств некоторого множества из N элементов?

Подвопросы:

- Какова связь между количеством элементов N в исходном множестве и количеством всех подмножеств?
- Для каких значений N требуемые подмножества можно получить в разумные сроки?
- Как называется техника поиска решений задач, которая основана на исследовании всего пространства всевозможных решений?

Субъективные итемы или итемы с полученными, составленными, разработанными или созданными учениками ответами – это задания, требующие результатов в виде интегрированных структур. Основными подтипами субъективных итемов являются: итемы с кратким ответом, итемы с развёрнутым ответом, итемы типа решение задач, итемы типа написание эссе и т. п.

Итемы с кратким ответом требуют от ученика выразить важную идею в нескольких предложениях: объяснение, отношение, описательное определение, элементарный алгоритм, математическая модель и т. п. Учитель определяет и сообщает ученикам однозначные критерии, в соответствии с которыми будут оцениваться ожидаемые ответы.

Пример.

Дан фрагмент программы P4 на языке PASCAL, в котором удалены имена типов используемых переменных:

Program P4;

var

q : _____;

s : **set of** _____;

i : _____;

begin

q:=‘069188288311’;

s:=[];

for i:=1 **to** length(q) **do**

s:=s+[ord(q[i])-ord(‘0’)];

...

end.

Проанализируйте фрагмент программы P4 и выполните следующие задания:

a) Впишите непосредственно в тексте программы пропущенные имена типов переменных таким образом, чтобы они не нарушали корректность операторов программы.

b) Запишите значение, которое получит переменная s после выполнения цикла **for**.

s = [_____]

Итемы типа решение задач. Процесс решения задачи – это вовлечение учащихся в новый вид учебной деятельности, отличающийся от ранее выполняемых ими. Основной целью итемов типа решение задачи является развитие креативности, дивергентного мышления, алгоритмического мышления, воображения, умения обобщать, переформулировать задачу и т.д. В зависимости от предложенной для решения задачи, с помощью итемов рассматриваемого типа проверяются компоненты компетенций, сформированных на этапах фиксации, интериоризации и экстериоризации. Решение задачи с помощью компьютера подразумевает выполнение ряда определенных этапов, которые могут оцениваться и отдельно, с целью более точного определения прогресса каждого из учеников.

Пример.

Банковские карты

С каждой банковской картой ассоциированы два цифровых кода.

Основной код, обозначаемый как *PIN* (*персональный идентификационный номер*), представляет собой четырехзначное число. В тех случаях, когда пользователь несколько раз подряд вводит числа, не совпадающие с *PIN*-кодом, банкомат блокирует доступ к карте.

Второй код, обозначаемый как *PUK* (*персональный ключ разблокировки*) представляет собой трехзначное число. Он используется банкоматом для разблокировки доступа к карте.

Одним из критериев принятия банкоматом введенного пользователем числа в качестве *PUK*-кода состоит в том, что сумма цифр введенного числа является одним из делителей суммы цифр *PIN*-кода соответствующей карты.

Задание. Напишите программу, которая для данного *PIN*-кода проверяет, соответствует ли номер введенный пользователем в качестве *PUK*-кода критерию его принятия. Программа должна содержать подпрограмму, которая принимает в качестве входного параметра целое число *N* и возвращает сумму его цифр *S*.

Ввод. Текстовый файл *CODE.IN* содержит в первой строке код *PUK* – целое число без знака, состоящее из трех цифр. Вторая строка содержит *PIN*-код – целое число без знака, состоящее из четырех цифр.

Вывод. Текстовый файл *CODE.OUT* должен содержать в единственной строке слово ПРИНЯТО если введенный *PUK*-код соответствует критерию его принятия. В противном случае, выходной файл должен содержать слово ОТКАЗАНО.

Итемы типа эссе позволяют проводить всеобъемлющее оценивание уровня развития тех компетенций, которые не могут быть оценены в полном объеме с помощью объективных или полубъективных итемов. С помощью итемов рассматриваемого типа оцениваются навыки организации и интеграции идей, способность изложения собственных мыслей в письменной форме, умение интерпретировать и использовать полученные данные. В случае итемов типа эссе ученик должен написать в свободном изложении требуемый ответ. На уроках Информатики можно предлагать учащимся написание структурированных или полуструктурированных мини-эссе с ожидаемыми, упорядоченными и направленными

ответами на следующие темы: структура компьютера, топология компьютерных сетей, интернет-сервисы, безопасность в Интернете и т. п.

Примеры:

- Опишите преимущества и недостатки метода перебора.
- Сравните эффективность методов сортировки, например, алгоритма сортировки выбором и алгоритма пузырьковой сортировки.

Итeмы с развернутым ответом предполагают расширенный объем ответа и в случае школьной дисциплины *Информатика* обычно не используются.

Виды текущего оценивания. Рекомендуется использование следующих видов текущего оценивания: наблюдение, устный опрос, выполнение упражнений и решение задач, стандартизированные тесты. Ниже приведено краткое описание каждого из этих видов оценивания.

Текущее наблюдение школьного поведения ученика может осуществляться практически на любых этапах учебной деятельности. Хотя этот способ иногда применяется и для оценивания результатов учащихся, чаще всего он используется для оценивания аффективно-поведенческих характеристик учеников.

Характеристики, которые возможно оценивать:

На уровне концепций и способностей:

- организация и интерпретация данных;
- подбор и правильная организация рабочих инструментов;
- описание и обобщение определенных процессов, приемов, взаимосвязей;
- использование дополнительных материалов в процессе доказательства;
- выявление отношений;
- использование компьютеров в соответствии со спецификой решаемых задач.

На уровне отношений:

- сосредоточенность на решаемой задаче;
- активное участие в решении поставленной задачи;
- формулирование вопросов по теме;
- завершение выполнения задания;
- обзор используемых методов и результатов.

Устный опрос. Является видом индивидуального экспресс-оценивания, который относится к традиционным видам и считается малоэффективным в образовании, ориентированном на ученика, в образовании, в котором дидактическая дискуссия с участием большого числа учащихся имеет большую эффективность. Как правило, для оценивания учеников, участвующих в дискуссии, применяется метод наблюдения.

Выполнение упражнений и решение задач. Целью рассматриваемой образовательной деятельности является усвоение и закрепление полученных новых знаний. Результат деятельности ученика оценивается заключением типа правильно/неправильно. Кроме того, проводится и оценивание самого процесса получения результата, определяются «сильные и слабые стороны» каждого из этапов выполнения упражнения или получения решения. Упражнения и задачи могут быть

традиционными (выполняются в письменном виде), так и инновационными (выполняются на компьютерах).

Стандартизированные тесты. Письменные или компьютерные тесты (локальные или на специализированных серверах) следует использовать выборочно или фронтально на заключительных этапах урока или другой краткосрочной учебной деятельности. Как правило, такие тесты должны включать итемы объективного и полу-объективного типов, измеряющие уровень усвоения только тех знаний, которые были запланированы к усвоению именно в ходе соответствующей учебной деятельности. Стандартизированные тесты рекомендуется разрабатывать и администрировать с помощью специализированных приложений (например, Moodle, eFront, цифровые формуляры или электронные таблицы), что позволяет проводить оценивание сразу после завершения процесса тестирования. В Информатике, разработку и администрирование стандартизированных тестов следует проводить именно с помощью специализированных приложений, поскольку они обеспечивают большую диверсификацию типологии используемых итемов, позволяют автоматизировать процесс сбора и обработки статистических данных и, что особенно важно, дают возможность персонализировать и индивидуализировать тесты путем создания банков итемов.

Приложения

Приложение 1. Пример долгосрочного дидактического проекта. 10-й класс, реальный профиль

Авторы: **Светлана Голубев**, преподаватель Информатики,
высшая дидактическая степень;

Киструга Георге, преподаватель Информатики,
высшая дидактическая степень

Внимание! Преподаватели должны персонализировать долгосрочные дидактические проекты в соответствии от спецификой класса и доступных образовательных ресурсов согласно Куррикулуму по Информатике, издание 2019 года.

Специфические компетенции дисциплины:

- CS1. Использование цифровых инструментов для повышения эффективности процесса обучения и работы, демонстрируя инновационный и практический подходы.
- CS2. Взаимодействие с членами виртуальных сообществ в целях обучения и работы, проявляя интерес к активному обучению, исследованиям и сотрудничеству, соблюдая этику виртуальных сред.
- CS3. Распространение в цифровых средах результатов личных и командных достижений, доказывая изобретательность, командный дух и убежденность.
- CS4. Создание цифровых графических, видео и аудио продуктов, демонстрируя креативность и уважение по отношению к национальным и универсальным ценностям.
- CS5. Научное восприятие роли и воздействия информационных явлений на современное общество, проявляя критическое и позитивное мышление при интеграции различных областей познания и человеческой деятельности.
- CS6. Обработка данных экспериментов из области реальных и общественных наук, демонстрируя критическое мышление, ясность и корректность.
- CS7. Алгоритмизация методов анализа, синтеза и нахождения решений проблемных ситуаций, демонстрируя креативность и настойчивость.
- CS8. Внедрение алгоритмов с помощью сред разработки программ, демонстрируя сосредоточенность и настойчивость.
- CS9. Исследование проблемных ситуаций с помощью моделирования, планирования и проведения виртуальных экспериментов в цифровых средах, демонстрируя аналитический подход, ясность и лаконичность.

Библиография/Ресурсы

1. *Национальный Куррикулум. Дисциплина Информатика. X-XII классы.* Утвержден Национальным советом по куррикулуму (протокол № 22 от 05 июля 2019 г.).
2. *Методический гид к куррикулуму по Информатике (2019).*
3. Гремальски А., Мокану Ю., Спинеи И. *Информатика. Учебник для 9-го класса.* Știința, Chișinău, 2016.

4. Гремальски А., Брайков А., Корлат С. *Информатика. Учебник для 12-го класса. Știința, Chișinău, 2015.*
5. Gremalschi L., Andronic V, Ciobanu I. Chistruga Gh. *Informatica. Ghid de implementare a curriculumului modernizat pentru treapta gimnazială.* Lyceum, Chișinău, 2011.
6. Corlat S., Ivanov L., Bîrsan V. *Informatica. Ghid de implementare a curriculumului modernizat pentru treapta liceală.* Cartier, Chișinău, 2010.
7. Gremalschi A., Ciobanu I. *Informatică. Clasa a 9-a. Ghidul profesorului.* Știința, Chișinău, 2011.
8. Gremalschi A., Ciobanu I., Ivanov L., Prisăcaru A. *Referențial de evaluare. Disciplina Informatica // Referențialul de evaluare a competențelor specifice formate elevilor,* Institutul de Științe ale Educației, Chisinau, 2014.
9. *Standarde de competențe digitale ale elevilor din ciclul primar, gimnazial și liceal.* Aprobate prin ordinul Ministerului Educației nr. 862 din 7 septembrie 2015.
10. *Matematică și științe. Ghiduri metodologice. (Dezvoltarea și implementarea curriculumului în învățământul gimnazial).* Grupul Editorial Litera, Chișinău, 2000.
11. *Laborator virtual de Informatică și TIC.* <http://lab.infobits.ro/>.
12. <https://sites.google.com/a/para.ro/teste/Home/clasa-a-ix-a/limbajul-c>.
13. *Subiecte Olimpiada Republicană la Informatică 2018.* <http://aee.edu.md/content/ol18-teste-informatic%C4%83>.
14. <http://campion.edu.ro/rules.php>.
15. <https://infoarena.ro/downloads>.
16. <http://www.ctice.md/ctice2013/wp-content/uploads/2014/09/E.pdf>.

Администрирование дисциплины

Количество часов в неделю	Количество часов в учебном году
2	68

Учебные единицы	Кол-во часов	Из них, для оценивания
Правила техники безопасности и поведения в лаборатории Информатики	2	1
I. Методы описания естественных и формальных языков	4	1
II. Словарь и синтаксис языка программирования высокого уровня	6	
III. Концепция данных. Простые типы данных	10	1
IV. Концепция действия. Операторы языка программирования высокого уровня	8	
<i>Всего I семестр</i>	<i>30</i>	<i>3</i>
IV. Концепция действия. Операторы языка программирования высокого уровня (продолжение)	16	1
V. Модуль по выбору. Веб-дизайн	20	1
На усмотрение преподавателя	2	
<i>Всего II семестр</i>	<i>38</i>	<i>2</i>
Всего в учебном году	68	5

Дидактическое проектирование учебных единиц

Единицы компетенций	Учебные единицы/ Тематическое содержание	Распределение во времени		Дидактические стратегии (организационные формы, дидактические ресурсы, оценивание)	Примечания
		Кол-во часов	Дата		
<ul style="list-style-type: none"> - Соблюдение правил поведения и безопасности в кабинете Информатики. - Соблюдение правил гигиены труда на уроках Информатики. 	<p>Нормы техники безопасности в кабинете Информатики.</p> <p>Навыки применения компьютера.</p> <p>Начальное тестирование</p>	1		<ul style="list-style-type: none"> - Фронтальная деятельность. - Индивидуальная деятельность. - Тест. 	
		1			
1. Методы описания естественных и формальных языков – 4 часа					
<ul style="list-style-type: none"> - Объяснение способа описания грамматических конструкций через чередование, конкатенации, повторение и возможного включения. - Использование формул БНФ и синтаксических диаграмм для проверки корректности текста. 	<p>1.1. Способы описания грамматических конструкций. Метаязык БНФ.</p> <p>1.2. Способы описания грамматических конструкций. Синтаксические диаграммы.</p>	2		<ul style="list-style-type: none"> - Фронтальная деятельность. - Индивидуальная деятельность. - Упражнение. - Тематическое исследование. 	
		2			
2. Словарь и синтаксис языка программирования высокого уровня – 6 часов					
<ul style="list-style-type: none"> - Идентификация лексических единиц языка программирования высокого уровня. - Использование формул БНФ и синтаксических диаграмм для проверки корректности текста и лексических единиц. 	<p>2.1. Лексические единицы языка программирования высокого уровня. Алфавит языка. Словарь языка.</p> <p>2.2. Лексические единицы: специальные символы; ключевые слова.</p> <p>2.3. Лексические единицы: идентификаторы; разделители.</p>	1		<ul style="list-style-type: none"> - Фронтальная деятельность. - Индивидуальная деятельность. - Упражнение. - Тематическое исследование. - Демонстрация. - Наблюдение. - Компьютерное оценивание. 	
		2			

- Применение правил составления идентификаторов, строк символов, чисел, комментариев.	2.4. Лексические единицы: числа; строки символов; метки.	2		
	2.5. Суммативное оценивание (учебные единицы I и II).	1		
3. Концепция данных. Простые типы данных – 10 часов				
- Классификация простых типов данных на предопределённые и перечисляемые пользователи порядковые, непорядковые, анонимные и явно определённые типы. - Классификация данных программы на константы и переменные. - Использование идентифицируемых и совместимых типов; - Использование синтаксических диаграмм и металингвистических формул для объявления типов, констант и переменных.	3.1. Концепция данных. Определённые типы данных: целые.	1		- Фронтальная деятельность. - Индивидуальная деятельность. - Упражнение. - Решение задач
	3.2. Определённые типы данных: вещественные.	1		- Деятельность в группе.
	3.3. Определённые типы данных: логические.	1		- Практическая работа.
	3.3. Определённые типы данных: символьные.	1		- Тематическое исследование. - Демонстрация. - Наблюдение.
	3.4. Определённые типы данных: порядковые.	1		- Тест.
	3.5. Идентичные и совместимые типы.	1		
	3.6. Порядковые типы данных. Интервальные типы данных. Анонимные и явно определённые типы данных*.	2		
	3.7. Объявление: констант; типов данных. Описание переменных.	1		
3.8. Суммативное оценивание.	1			

4. Концепция действия. Операторы языка программирования высокого уровня – 24 часа				
<ul style="list-style-type: none"> - Идентификация составных частей программ; - Использование металингвистических формул и синтаксических диаграмм изучаемого оператора для проверки синтаксической правильности разрабатываемых программ. - Разработка программ, в которых используются стандартные процедуры чтения /записи. - Разработка программ, в которых используются изучаемые операторы. - Описание известных алгоритмов математических, физических, химических, биологических процессов и явлений, используя различные способы описания алгоритмов. - Разработка алгоритмов для решения задач из различных областей человеческой деятельности. - Перевод разработанных алгоритмов в программы. - Тестирование программ, интерпретация и анализ их результатов. 	4.1. Концепция действия. Понятие оператора.	1		<ul style="list-style-type: none"> - Фронтальная деятельность. - Индивидуальная деятельность. - Упражнение. - Решение задач. - Разработка программ. - Отладка программ. - Деятельность в группе. - Практическая работа. - Тематическое исследование. - Демонстрация. - Наблюдение. - Тест.
	4.2. Вывод алфавитно-цифровой информации на экран.	1		
	4.3. Ввод данных с клавиатуры.	1		
	4.4. Выражения. Вычисление выражений.	2		
	4.5. Программирование выражений. Разработка и отладка программ.	2		
	4.6. Операторы: вызова процедуры; пустой оператор; составной оператор; оператор присваивания.	2		
	4.7. Условный оператор. Разработка и отладка программ.	2		
	4.8. Оператор выбора. Разработка и отладка программ.	2		
	4.9. Оператор цикла со счетчиком. Разработка и отладка программ.	2		
	4.10. Оператор цикла с условием. Разработка и отладка программ.	2		
	4.11. Оператор цикла с постусловием. Разработка и отладка программ.	2		
	4.12. Суммативное оценивание.	1		

5-А. ВЕБ-ДИЗАЙН – 20 часов

<ul style="list-style-type: none"> - Идентификация требований и рекомендаций относительно Веб- документов. - Прохождение этапов разработки Веб- документа. - Разработка Веб- документов с помощью офисных приложений. - Разработка Веб- документов с помощью специализированных приложений. - Разработка Веб- документов с помощью онлайн приложений; - Публикация Веб- документов в среде Интернет. - Соблюдение национального и международного законодательства в области авторских прав. 	<p>5.1. Веб-документы: Понятия и концепты. Форматы Веб- документов. Структура сайта.</p> <p>5.2. Требования к Веб- документам. Этапы разработки Веб- документов. Программные приложения для создания и генерации Веб- документов..</p> <p>5.3. Объекты из состава документов Веб: текст.</p> <p>5.4. Объекты из состава документов Веб: списки.</p> <p>5.5. Объекты из состава документов Веб: таблицы.</p> <p>5.6. Объекты из состава документов Веб: изображения.</p> <p>5.7. Объекты из состава документов Веб: ссылки.</p> <p>5.8. Создание документов Веб при помощи офисных приложений.</p> <p>5.9. Создание документов Веб при помощи специальных приложений.</p>	<p>1</p> <p>1</p> <p>1</p> <p>1</p> <p>2</p> <p>2</p> <p>1</p> <p>2</p> <p>3</p>	<ul style="list-style-type: none"> - Фронтальная деятельность. - Индивидуальная деятельность. - Упражнение. - Решение задач. - Деятельность в группе. - Практическая работа. - Тематическое исследование. - Проект. - Демонстрация. - Наблюдение. - Тест.
--	--	--	--

	5.10. Создание документов Веб при помощи онлайн приложений.	3		
	5.11. Публикация документов Веб в Интернете.	1		
	5.12. Элементы права: авторские права; смежные права; виды лицензий (ССЛ); анти-плагиат в области мультимедиа.	1		
	5.13. Суммативное оценивание.	1		
По усмотрению преподавателя		2		

Приложение 2. Пример краткосрочного проекта (проект урока). 10-й класс, гуманитарный профиль

Информатика, X класс, гуманитарный профиль

ДИДАКТИЧЕСКИЙ ПРОЕКТ № 2

Киструга Георге, учитель информатики, высшая дидактическая степень

Общие данные. Календарная дата, 10-й класс, гуманитарный профиль, информатика.

Модуль: Методы описания естественных и формальных языков.

Тема урока. Методы описания грамматических конструкций. Синтаксические диаграммы.

Тип урока: Приобретение и применение знаний.

Единицы компетенций:

1. Описание грамматических конструкций через альтернативу, соединение, повторение и опциональное включение.
2. Применение синтаксических диаграмм для проверки правильности текста и лексических единиц.

Операционные цели.

По окончании урока ученики смогут:

- O₁ – Идентифицировать элементы синтаксических диаграмм и объяснять их назначение.
- O₂ – Составлять грамматические конструкции на основе предложенных синтаксических диаграмм.
- O₃ – Рисовать синтаксические диаграммы на основе предложенных BNF формул.
- O₄ – Составлять BNF формулы на основе предложенных синтаксических диаграмм.
- O₅ – Проявлять интерес и настойчивость при выполнении предложенных заданий.

Дидактические стратегии:

- Формы организации: фронтальная, индивидуальная.
- Дидактические методы, процедуры и техники: беседа, изложение, анализ, проблематизация.
- Дидактические средства: учебник по Информатике, интерактивная доска; карточки; компьютеры.

Стратегии оценивания: формативное интерактивное оценивание.

Место проведения: кабинет информатики.

Библиография:

1. *Национальный Куррикулум. Дисциплина Информатика. X-XII классы.* Утвержден Национальным Советом по Куррикулуму (протокол № 22 от 05 июля 2019 г.).

2. *Методический гид к Куррикулуму по Информатике (2019).*
3. Gremalschi A., Ciobanu I., Ivanov L., Prisăcaru A. *Referențial de evaluare. Disciplina Informatica // Referențialul de evaluare a competențelor specifice formate elevilor*, Institutul de Științe ale Educației, Chisinau, 2014.
4. Гремальски А., Мокану Ю., Спирей И. *Информатика. Учебник для IX класса, Știința*, Chisinau, 2016.

ХОД УРОКА

I. ВЫЗОВ (5-10 минут)

1. **Организация класса.** Приветствие учеников (вначале ученики приветствуют учителя, затем друг друга).
2. **Концентрация внимания:**
 - а) Рассмотрим следующие формы представления информации, используемые в Античном Египте:

С помощью иероглифов



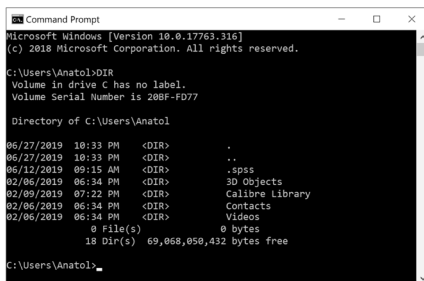
С помощью рисунков



Какое из этих представлений более интуитивно? Аргументируйте ваш ответ.

- б) Известно, что взаимодействие человека и компьютера осуществляется с использованием следующих типов интерфейсов:

Интерфейс командной строки



Графический интерфейс



Который из этих интерфейсов проще и удобней для работы? Аргументируйте ваш ответ.

Вывод: В определённых случаях, графическое представление информации является более интуитивным и удобным для пользователя.

Исходя из этого, в Информатике, для описания грамматических конструкций, помимо текстов (БНФ формул), используются специальные графические представления, называемые синтаксическими диаграммами.

II. ОСМЫСЛЕНИЕ (20-30 минут)

3. Сообщение темы и целей урока словами, понятными ученикам.

Преподаватель:

- Объявляет тему урока и записывает на доске: Методы описания грамматических конструкций. Синтаксические диаграммы.
- Объявляет цели урока.
- Представляет информацию по теме урока.

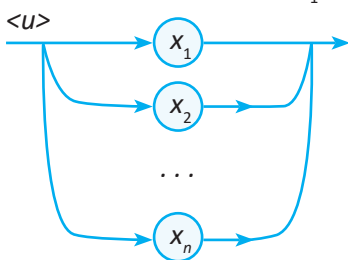
Синтаксические диаграммы более наглядно описывают синтаксис языков программирования. Диаграммы может быть получены из БНФ формулах следующим образом:

Каждому терминальному символу БНФ формулы соответствует круг или овал, в который вписывается соответствующий символ. Нетерминальные символы вписываются в прямоугольники. Овалы и прямоугольники соединяются в соответствии со следующими рисункам:

Конкатенация: $\langle u \rangle ::= x_1 x_2 \dots x_n$



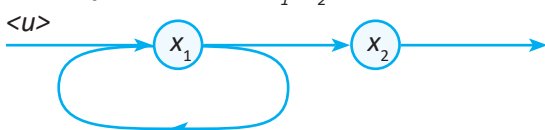
Альтернатива: $\langle u \rangle ::= x_1 / x_2 / \dots / x_n$



Присутствие необязательно: $\langle u \rangle ::= [x_1] x_2$



Повторение: $\langle u \rangle ::= \{x_1\} x_2$



Деятельность учеников: Ученики записывают в тетрадях содержание нового урока. Следят за объяснениями.

4. Исследование представленной информации, наблюдение и рефлексия.

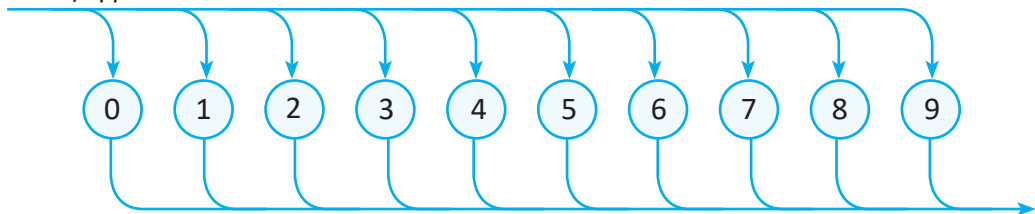
а) Ученикам предлагается набросать диаграммы, аналогичные представленным выше, выделяя их свойства и уточняя способы их рисования.

<День> ::= Понедельник | Вторник | Среда | Четверг | Пятница | Суббота | Воскресенье

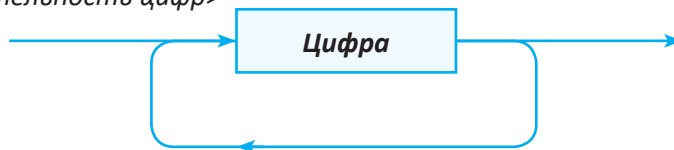


б) Ученикам предлагается написать БНФ формулы, которые соответствуют синтаксическим диаграммам приведенном ниже, и дать примеры дробных чисел.

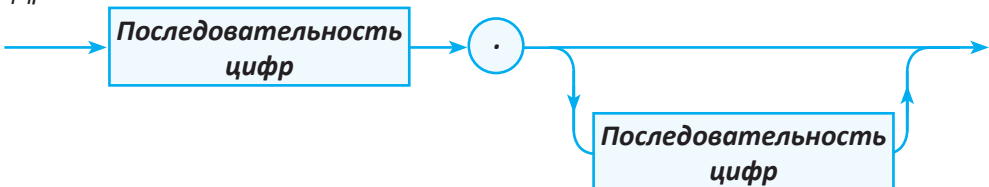
<Цифра>



<Последовательность цифр>



<Дробное число>



Ответ:

<Цифра> ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

<Последовательность цифр> ::= < Цифра>{< Цифра>}

<Дробное число> ::= < Последовательность цифр>.[< Последовательность цифр>]

Примеры дробных чисел: 23.411; 0.28; 3.0001; 00.0001; 000.12009.

III. РЕФЛЕКСИЯ (5-7 минут)

Презентация/усвоение конкретного опыта. Ученикам предлагаются несколько заданий, решение которых требует использование изученной/ известной информации.

а) Индивидуальная деятельность. Нарисуйте синтаксические диаграммы, соответствующие нижеприведенным БНФ формулам:

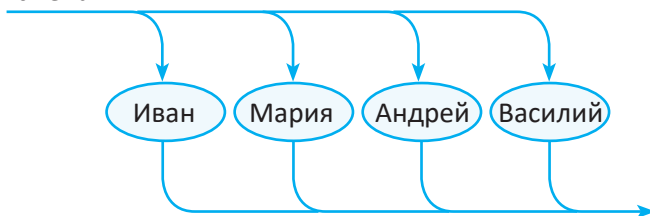
<Ученик> ::= Иван | Мария | Андрей | Василий

<Цифровое устройство> ::= смартфон | планшет

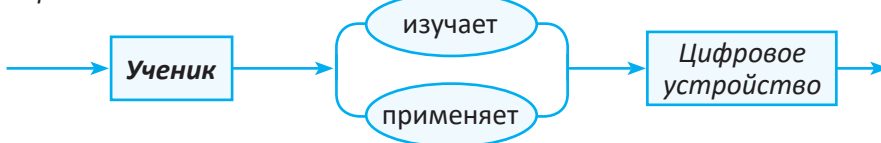
<Предложение> ::= < Ученик>изучает | применяет< Цифровое устройство>.

Ответ:

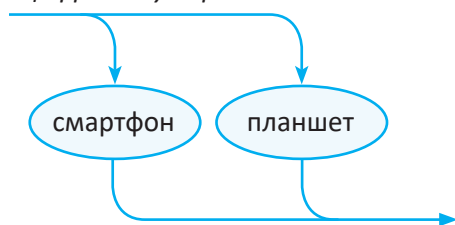
<Ученик>



<Предложение>



<Цифровое устройство>



б) Групповая деятельность. Учитель просит каждую из групп написать несколько предложений, составленных в соответствии с синтаксической диаграммой <Предложение>.

Возможные ответы:

Иван изучает смартфон.

Мария применяет планшет.

Андрей изучает планшет.

- с) **Групповая деятельность.** Ученикам предлагается найти число всевозможных предложений, которые можно составить в соответствии с синтаксической диаграммой <Предложение>.

Ответ: 16.

5. Оценивание урока. Самооценивание, рефлексия.

В конце урока ученикам предлагается сформулировать выводы, выразить свои эмоции, переживания, впечатления, накопленные в процессе проведения урока, сообщая:

- Самое любопытное, о чем они узнали на уроке.
- Самое интересное из полученной на уроке информации.
- Пример информации, которую они знали до начала урока.
- Информацию, которую они так и не поняли.
- Запомнившийся курьез.
- Вопрос.
- Просьба/запрос.

IV. РАСШИРЕНИЕ (2-4 минуты)

6. Домашнее задание:

- а) Повторите параграф «Мета-язык БНФ» из учебника.
- б) Изучите параграф «Синтаксические диаграммы» из учебника.
- с) Выполните упражнения 4 и 6 из параграфа «Синтаксические диаграммы» из учебника.

Приложение 3. Пример проблемного обучения. Обработка двумерных таблиц

Проблема. Чтобы найти районы с наибольшей опасностью затопления, агентство *Apele Moldovei* провело картографирование поймы Прута. С этой целью, карта поймы была разделена на квадратные участки со стороной 500 метров (рисунок 1). Квадратный участок считается «с риском затопления», если через него проходит русло реки. В противном случае, квадратный участок считается «безопасным». Агентству необходимо определить общую площадь участков с риском затопления.

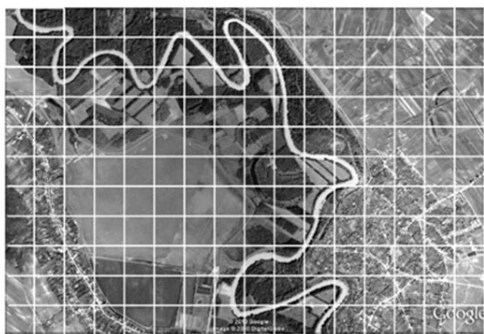


Рисунок 1

1	1	1	0	0	1	1	1	1	0	0	0	0	0	0	0
0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0
0	1	1	0	1	1	0	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	1	0	1	1	1	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0

Рисунок 2

1	1	1	0	0	1	1	1	1	0	0	0	0	0	0	0
0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0
0	1	1	0	1	1	0	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	1	0	1	1	1	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0

Рисунок 3

Поскольку объем цифровой карты поймы Прута очень велик, специалисты агентства представили единицами (1) участки с риском затопления и нулями (0) безопасные участки и сохранили полученную таким образом таблицу (рисунок 3). В учебных целях, предполагается, что число участков как по горизонтали, так и по вертикали не превышает 20.

Обсуждение:

1. Как можно решить задачу?

Ответ: Необходимо найти число участков с риском наводнений, а затем умножить их число на площадь одного участка.

2. Можно ли решить задачу вручную?

Ответ: Да.

3. А если таблица, представляющая карту поймы будет иметь 100 строк и 1000 столбцов? А если потребуется выполнить картографирование пойм других рек?

Ответ: Следует написать компьютерную программу.

4. Как можно представить карту в компьютере?

Ответ: Следует использовать двухмерный массив.

5. Как ввести исходные данные?

Ответ: С клавиатуры или путем чтения из текстового файла.

Примечание. Для приведенного выше примера объем вводимых данных достаточно велик, поэтому рассмотрение данной задачи является более целесообразным после изучения учениками текстовых файлов.

6. Как вычислить число участков с риском затопления?

Ответ: Путем построчного просмотра всех элементов таблицы, от первого до последнего столбца, и подсчете элементов со значением «1». Очевидно, до начала просмотра элементов таблицы, число участков с риском затопления считается равным 0.

7. Как рассчитывается общую площадь участков с риском затопления?

Ответ: Пусть K – количество участков с риском затопления; L – длина стороны участка в км; S – общая площадь участков с риском затопления. Очевидно, $S = K \times L \times L \text{ км}^2$.)

Алгоритм. Предположим, что карта поймы Прута имеет M квадратных участков по длине и N участков по высоте. Сторона квадратного участка имеет длину L . Данные вводятся в двумерный массив A с N строками и M столбцами.

Предварительные условия: Необходимо ввести значения N , M , L и сами значения элементов массива A .

Шаг 1. $K \leftarrow 0$.

Шаг 2. Пусть I – индекс текущей строки. $I \leftarrow 1$.

Шаг 3. Для всех J от 1 до M повторяем: если $A[I, J]$ равно 1, то $K \leftarrow K + 1$.

Шаг 4. Если $I < N$ (еще не просмотрены все строки), то $I \leftarrow I + 1$ и возвращаемся к Шагу 3. В противном случае переходим к Шагу 5.

Шаг 5. $S \leftarrow K \times L \times L$.

Шаг 6. Выводим S . Конец.

Реализация алгоритма на языке программирования высокого уровня не представляет трудностей и остается на усмотрение читателя.

Обзор деятельности по решению задачи.

1. Ученики обсуждают реальную ситуацию (проблему).
2. Ученики выполняют процесс абстрагирования проблемы, выявляя связи между реальными объектами и абстрактной моделью, которая их представляет.
3. Ученики строят алгоритм решения.
4. Ученики проверяют полученные результаты, обсуждают их, предлагают аналогичные ситуации, которые сводятся к этой же проблеме.
5. На каждом из перечисленных выше этапов, преподаватель консультирует учеников и, в случае необходимости, направляет их с помощью наводящих вопросов или ориентировочных указаний.

Приложение 4. Пример проектного обучения. Эффективность алгоритмов сортировки

Ученикам проект может быть предложен с различными уровнями сложности, в зависимости от алгоритмов сортировки, выбранных для сравнительного анализа.

Для проекта с низкой сложностью, могут быть предложены методы сортировки с помощью пузырьков, путем выбора или вставки. Проект средней сложности включает сравнение медленных алгоритмов (метод пузырьков метод, сортировка путем выбора или вставки) с быстрым алгоритмом (быстрая сортировка или *heapsort*). Повышенная сложность проекта достигается при анализе эффективности оптимальных алгоритмов: быстрая сортировка и *heapsort* или быстрая сортировка и *mergesort*.

Для выполнения проекта, ученику (или группе учеников) потребуются следующие ресурсы:

- Информационные: описание алгоритмов выбранных методов; описание способов измерения временной сложности алгоритмов; метод присвоения случайных значений элементам массива.
- Оборудование: компьютер, доступ в интернет, мультимедийный проектор.
- Компьютерные программы: среды разработки программ на языках программирования высокого уровня FreePascal, Dev C ++ или аналогичные.

Предполагаемая продолжительность проекта: от 7 до 14 дней.

Сроки реализации:

- после изучения единицы обучения “Анализ алгоритмов” (начальный уровень);
- после изучения единицы обучения „Метод Разделяй и властвуй” (средний уровень);
- не фиксируется (продвинутый уровень).

Этапы проекта:

Этап 1 (реализуется преподавателем): выбор темы, ресурсов, составление рекомендаций для групп из учеников, работающих над проектом, составление предварительного графика.

Этап 2 (день 1): Формирование групп, формулировка цели проекта, распределение обязанностей, объяснение понятий, требований, методов выполнения проектов.

Этап 3 (день 8): Ответы на вопросы, анализ предложений учеников, выдача советов, объяснения, проверка текущего состояния дел.

Этап 4 (день 15): презентация проекта, оценивание, обсуждение в классе.

Далее в качестве примера представлен проект по анализу эффективности алгоритмов сортировки с использованием метода пузырьков и метода быстрой сортировки.

Цель исследования: Анализ эффективности алгоритмов сортировки методом пузырьков и методом быстрой сортировки в случае массивов больших размеров. Выработка рекомендаций по использованию методов сортировки.

Описание источников для документирования и ресурсов для программирования. Для выполнения проекта следует изучить алгоритм сортировки с использованием метода пузырьков (<http://en.wikipedia.org/wiki/Bubblesort>) и алгоритм быстрой сортировки (<http://en.wikipedia.org/wiki/Quicksort>). Для разработки программ более предпочтительным является бесплатный компилятор Free Pascal, который с помощью стандартных операций позволяет обрабатывать линейные массивы больших размера.

Описание метода исследования. Сравнение времен сортировки каждого из предложенных преподавателем алгоритмов сортировки.

С этой целью каждый из исследуемых алгоритмов программируется в виде отдельной процедуры. Далее, создаются массивы с заранее определенным количеством элементов, значения которых задаются случайным образом. Созданные массивы сортируются с помощью процедур, описанных выше, измеряя при этом время выполнения каждой из них.

Программа на языке ПАСКАЛЬ, реализующая описанный выше метод исследования, имеет вид:

```
program p001;
uses crt, dos;
const n=30000;
type t=array[1..n] of integer;
var a,b: t;
    t1,t2: real;
    i:integer;

procedure bubble; {сортировка пузырьковым методом}
var i,j,tmp: integer;
begin for i:=1 to n-1 do
      for j:=1 to n-i do
        if a[j]>a[j+1] then begin
                               tmp:=a[j]; a[j]:=a[j+1]; a[j+1]:=tmp;
        end;
      end;
end;

procedure qsort(st,dr:integer); {Быстрая сортировка}
var s,i,k,tmp : integer;
begin
if (st < dr) then begin
    s:=b[st]; k:=st;
    for i:=st+1 to dr do
      if b[i]<s then begin k:=k+1; tmp:=b[i];
                        b[i]:=b[k]; b[k]:=tmp;
                      end;
    b[st]:=b[k]; b[k]:=s;
    qsort(st,k-1); qsort(k+1,dr);
end;
```



```

end
    else exit;
end;
function timpcurrent : real;
var h,m,s,ms: word;
begin
gettime(h,m,s,ms);
    timpcurrent:=h*3600+m*60+s+ms/100;
end;

begin
clrscr;
{Генерация случайных значений для элементов массива}
randomize;
for i:=1 to n do
begin
    a[i]:= 1+random(30000);
    b[i]:=a[i];
end;
t1:=timpcurrent; {фиксирование времени до начала вызова процедуры}
bubble;
t2:=timpcurrent; {фиксирование времени после вызова}
writeln('metoda bulelor ', n, ' elemente. timp: ', t2-t1:0:10);
t1:=timpcurrent;
qsort(1,n);
t2:=timpcurrent;
writeln('sortarea rapida ', n, ' elemente. timp: ', t2-t1:0:10);
end.

```

Анализ результатов измерений. Учитель рекомендует ученикам измерять длительность времени выполнения каждой из процедур в контексте увеличения количества элементов массивов, подлежащих сортировке. Напоминаем, что в случае метода пузырьков время выполнения увеличивается с увеличением количества отсортированных элементов, в то время как в случае метода быстрой сортировки время выполнения практически не увеличивается. Также, обращаем внимание на тот факт, что иногда в случае быстрой сортировки, время, необходимое для сортировки небольших наборов данных, может превышать время, необходимое для сортировки больших наборов.

Выводы, ожидаемые от учеников:

1. В общем, алгоритм быстрой сортировки более эффективен, чем алгоритм сортировки методом пузырьков.
2. Для данных, объем которых не превышают примерно тысячи элементов, тип используемого алгоритма не является существенным. В случае же больших наборов данных жестких временных ограничений предпочтительней является алгоритм быстрой сортировки.

3. Время, необходимое для выполнения быстрой сортировки, не обязательно растет пропорционально с ростом количества элементов. Кроме того, эффективность алгоритмов зависит от исходного расположения элементов сортируемых массивов.
4. В случае реальных задач, решение которых требует сортировку данных, следует использовать более быстрые алгоритмы, даже если для этого потребуются больше усилий по программированию. Примеры таких задач: сортировка данных из национальных регистров транспортных средств, регистров населения, журналов банковских транзакций и т.п.

Примечания:

1. Проект легко трансформируется в тематическое исследование, если ученикам в качестве ресурса предлагаются подпрограммы, которые выполняют алгоритмы сортировки.
2. Для публичного представления проекта учащиеся будут использовать мультимедийный проектор и систему для электронных презентаций (Power Point [MS Office], Impress [Open Office] или другие).

Приложение 5. Пример обучения на основе тематического исследования.

Публикация веб-документов

Тематическое исследование в рамках модуля «Веб-дизайн» может быть предложено как в общем виде, так и в виде индивидуальных вариантов:

- бесплатная публикация веб-документов;
- коммерческая публикация веб-документов;
- публикация веб-документов с использованием платформ Web 2.0.

Выбор варианта представляет собой первый этап тематического исследования (выбор случая).

Следующий шаг (запуск случая) начинается после того, как учащиеся создали веб-документы (с помощью офисных приложений или помощью HTML), используя наиболее подходящий дидактический метод, например, беседу.

Для исследования каждого из варианта могут быть сформированы одна или несколько групп.

Продолжительность тематического исследования составляет примерно одну неделю.

Ресурсы, необходимые для проведения тематического исследования:

- Компьютер с доступом в интернет.
- Мультимедийный проектор (для окончательной презентации).
- Приложения для поиска информации.
- Электронная энциклопедия www.wikipedia.org.
- Информационные ресурсы на сайте www.molddata.md (разделы доменов, хостинг, законодательство).
- Информационные ресурсы на сайте www.moldtelecom.md (раздел интернет, дополнительные услуги).
- Приложения для создания цифровых презентаций.
- Тематические библиографические ресурсы.

Цели тематического исследования

1. Анализ средств, предназначенных для публикации веб-документов.
2. Определение этапов публикации веб-документа.
3. Выявление преимуществ и недостатков для каждого из идентифицированных методов публикации веб-документов.
4. Презентация результатов исследования своим товарищам по классу.

План выполнения тематического исследования

День 1. Объяснение и распределение задач, подбор информационных ресурсов, установление режима взаимодействия членов рабочих групп на период выполнения тематического исследования.

Дни 2-7. Члены группы собирают, систематизируют, анализируют материалы, относящиеся к тематическому исследованию. Преподаватель дает онлайн или очные консультации членам рабочих групп.

День 8. Презентация результатов тематического исследования перед слушателями (классом). Ученики приглашаются разработать и доложить презентацию, участвовать в дебатах, делать выводы.

Пример тематического исследования, ожидаемого от учащихся

Введение. Пусть мы имеем сайт, который мы спроектировали, разработали, запрограммировали и проверили его функциональность. Очевидно, пока его можно использовать только на локальном компьютере. Ставится задача опубликовать его в Интернете, сделав его тем самым доступным для всех пользователей.

Для публикации сайта в Интернете необходимо выполнить следующие шаги:

- Назначить доменное имя.
- Выбрать веб-сервер на котором будет находиться сайт (хостинг).
- Организовать и дать имена файлам из состава сайт в соответствии с требованиями хост-сервера.
- Загрузить файлы на сервер.

Доменное имя. Идентичность веб-сайта в Интернете обеспечивается с помощью доменного имени. Доменное имя является частью URL-адреса каждой из страниц сайта, указывая тем самым их местонахождение в Интернете.

Основные домены Интернете относятся к разным типам: коммерческие (.com); образовательные (.edu); правительственные (.gov); поставщиков сетевых услуг (.net); некоммерческих организаций (.org); географические (.md, .ro, .uk) и т.д.

Сам домен и, конечно же, его имя, можно получить платно либо бесплатно.

Платные домены. В случае если владелец сайта – юридическое или физическое лицо – желает иметь стабильного и заслуживающего доверия присутствия в Интернете, платный домен является идеальным выбором. Регистрация домена не дорогая, но она гарантирует идентичность и авторитет сайта и его владельца.

Например, в случае доменов .md, которые регистрируются Государственным Предприятием MoldData, стоимость за один календарный год составляет около 500 леев (указана цена на момент проведения тематического исследования).

В общем случае, в мире существуют много коммерческих организаций, предоставляющих услуги по регистрации платных доменных имен, например www.host.md, www.10-domains.com.

В случае платного доменного имени, URL сайта будет иметь вид:

[http://www.\[выбраное доменное имя\].com](http://www.[выбраное доменное имя].com)

Бесплатные домены. Получить бесплатное доменное имя очень просто. Есть много компаний, которые предлагают такие домены и, кроме того, и хостинг для сайта. Для доменных имен, предлагаемых бесплатно, структура URL сайта имеет вид:

[http://www.\[имя компании которая предоставляет домен\].com/\[имя сайта\]](http://www.[имя компании которая предоставляет домен].com/[имя сайта])

или

[http://www.\[имя сайта\].\[имя компании которая предоставляет домен\].com](http://www.[имя сайта].[имя компании которая предоставляет домен].com)

Примеры: <http://sites.google.com/site/teoriagrafurilor>

<http://scorlat.blogspot.com>

Для сайтов юридических лиц такая структура не совсем приемлема. В то же время, для образовательных, личных сайтов и т.п., бесплатный домен, предоставляемый компанией, которая не размещает рекламные страницы, использование бесплатных имен является удобным и быстрым решением.

В настоящее время наиболее популярными являются бесплатные сервисы, предлагаемые WordPress, Blogspot, Google.

Веб-хостинг. Если был выбран бесплатный домен, услуги хостинга являются частью пакета хостинга, предлагаемого принимающей компанией (например, в настоящее время Google предлагает до 100 МБ для одного сайта, однако позволяет создавать и несколько связанных сайтов).

В случае коммерческих услуг, существует несколько вариантов хостинга:

- Установить свой собственный веб-сервер.
- Использовать веб-сервер Интернет-провайдера.
- Использовать веб-сервера, принадлежащий компании, предлагающей услуги хостинга.

Собственный веб-сервер. Это самое дорогое решение, но оно предлагает максимальную гибкость и свободу. Использование собственного сервера требует покупки специализированного оборудования, постоянного широкополосного доступа в Интернет, специальных услуг, предоставляемых поставщиком интернет-услуг и т.п. Собственный веб-сервер является оптимальным решением для крупных компаний с отделами, занимающимися предоставлением веб-сервисов.

Веб-сервер Интернет-провайдера. Использование веб-сервера Интернет-провайдера является рациональным и недорогим решением. Многие Интернет-провайдеры предоставляют дополнительно и услуги хостинга. Недостатком решения являются ограничения, налагаемые провайдером относительно выделенного пространства и самой структуры сайта.

Использование хостинг-сервера. Это оптимальное решение, поскольку оно позволяет устанавливать цену на услуги в соответствии с требуемым пространством и набором запрашиваемых услуг. Кроме того, компании предлагают различные возможности: динамические страницы, скрипты, базы данных, персонализированную электронную почту, динамическое расширение выделенного пространства и т.д. В Молдове такие услуги предлагаются, например, предприятием MoldData (через www.host.md).

Организация и именование файлов. Этот этап выполняется во время программирования сайта, но его правильное функционирование после публикации зависит от результатов проверки правильности имен папок и файлов, их взаимного размещения перед передачей.

Организационные рекомендации. Желательно, чтобы файлы на сайте были организованы на локальном компьютере точно так, как они будут организованы на веб-сервере. Веб-сервер имеет «корневой» каталог, в котором будут храниться все файлы сайта. Каталог, в котором вы будете хранить файлы на локальном компьютере, будет играть роль «корневого» каталога сервера. Файл, содержащий главную страницу веб-документа, будет помещен в корневой каталог.

Для небольших сайтов все файлы могут быть размещены в одном каталоге. Для более развитых структур рекомендуется разбивать разделы на отдельные каталоги, причем доступ к каждому разделу должен осуществляться через собственную главную страницу.

Для изображений рекомендуется создавать в «корневом» каталоге отдельный каталог, в котором следует разместить все изображения сайта. Пути от страницы, которая ссылается на эти изображения, должен быть одинаковыми как на локальном компьютере, так и на сервере.

Передача файлов. Операция переноса файлов, составляющих сайт, с локального компьютера на веб-сервер является относительно простой и состоит в копировании файлов на сервер на место, выделенное сайту. Некоторые хостинговые компании предоставляют специальные приложения для передачи файлов, но наиболее часто используемое средство передачи является клиентское приложение FTP.

Аннотация. На этапе веб-публикации сайта необходимо предпринять следующие шаги:

- Получить для сайта доменное имя, платное или бесплатное.
- Найти подходящий хостинг, исходя из соотношения цена/качество.
- Организовывать и именовать каталоги и файлы в соответствии с требованиями системы хост-сервера.
- Передать файлы на хост-сервер наиболее распространенным способом передачи – FTP.

Анализ

1. Тематическое исследование проводилось на основе достоверных данных, взятых с сайтов специализированных компаний в Республике Молдова.
2. Были определены все основные этапы публикации локально созданного сайта.
3. Проанализированы основные варианты выполнения каждого из этапов публикации сайтов.

Выводы:

1. Персональные тестовые сайты с образовательным контентом можно публиковать путем использования бесплатных сервисов.
2. Для коммерческих сайтов рекомендуются персонализированные платные домены.
3. Перед публикацией сайта следует выполнить проверку имен и локаций файлов.
4. Конкретный вид процедуры публикации сайта зависит, с одной стороны, от структуры и свойств самого сайта, и, с другой стороны, от специфики услуг хостинговой компании.

БИБЛИОГРАФИЯ

1. *Cadrul de referință al Curriculumului Național*. Aprobato prin ordinul Ministerului Educației, Culturii și Cercetării al Republicii Moldova nr. 432 din 29 mai 2017.
2. *Curriculumul actualizat la disciplina școlară «Informatica», învățământ gimnazial*. Aprobato prin ordinul Ministerului Educației nr. 936 din 28 august 2014 cu statut de document destinat desfășurării experimentului pedagogic.
3. *Curriculumul de bază. Sistem de competențe pentru învățământul general*. Aprobato la ședința Consiliului National pentru Curriculum din cadrul Ministerului Educației, Culturii și Cercetării al Republicii Moldova, proces-verbal nr. 1939 din 28 decembrie 2018.
4. *Evaluarea Curriculumului Educațional. Aria curriculară: Matematică și științe (studii curriculare)* / Achiri I., Bîrnaz N., Ciuvaga V. [et. al.]; coord.: Guțu VI. Universitatea de Stat din Moldova, UNICEF Moldova. – Chișinău: CEP USM, 2018.
5. *Standarde de competențe digitale ale elevilor din ciclul primar, gimnazial și liceal*. Aprobato prin ordinul Ministerului Educației nr. 862 din 7 septembrie 2015.
6. Cartaleanu T., Cosovan O., Goras-Postică V., et al., *Formare de competențe prin strategii didactice interactive*, Centrul Educațional Pro Didactica, Chișinău, 2008.
7. Corlat S., Ivanov L. *Calculatorul în predare și învățare. Ghid metodologic pentru formarea cadrelor didactice din învățământul preuniversitar*. I.E.P. Știința, Chișinău, 2007.
8. Gremalschi A., Prisacaru A. *Formarea și dezvoltarea culturii informaționale și a gândirii algoritmice în învățământul general* / Didactica Pro., nr. 4-5 (110-111), 2018.
9. Gremalschi A., Ciobanu I., Ivanov L., Prisăcaru A. *Referențial de evaluare. Disciplina Informatica // Referențialul de evaluare a competențelor specifice formate elevilor*, Institutul de Științe ale Educației, Chisinau, 2014.
10. Gremalschi A., Prisăcaru A. *Modernizarea curriculumul liceal la Informatică / Acta et commentationes*. Științe ale Educației, nr. 2 (9), 2016.
11. Guțu V., Chicu V., Dandara O. et al., *Psihopedagogia centrată pe copil*, Centrul Educațional-Poligrafic al USM, Chișinău, 2008.
12. Lupu I., Gremalschi A., Prisacaru A. *Dezvoltarea curriculară în informatică / Acta et commentationes*. Științe ale Educației, nr. 1(12), 2018.
13. *Национальный куррикулум. Дисциплина Информатика. X–XII классы*. Утвержден Национальным советом по куррикулуму (протокол № 22 от 05 июля 2019 г.).

14. *Обновленный куррикулум по предмету Информатика. X–XII классы.* Утверждён приказом Министерства просвещения Республики Молдова № 936 от 28 августа 2014 в качестве нормативного документа, предназначенного для проведения педагогического эксперимента.
15. *Communication from the Commission to the European Parliament, the Council, the European economic and social committee and the Committee of the regions on The Digital Education Action Plan.* Brussels, 17.1.2018. COM (2018) 22 final.
16. *Informatics Education in Europe: Are We All In The Same Boat?* ACM ISBN: #978-1-4503-5361-8.
17. Horst Shaub, Karl G. Zenke. *Dicționar de pedagogie*, Iași, Polirom, 2001.
18. Sue Sentance, Erik Barendsen, Carsten Schulte. *Computer Science Education / Perspectives on Teaching and Learning in School.* Bloomsbury Academic, 2018.
19. Marilyn Fryer. *Predarea și învățarea creativă.* Editura Uniunii Scriitorilor, Chișinău, 2004
20. Bocoș M., *Teoria și practica cercetării pedagogice*, Editura Casa Cărții de Știință, Cluj-Napoca, 2003.
21. Ioan Cerghit. *Sisteme de instruire alternative și complementare. Structuri, stiluri, strategii.* Polirom, 2008.
22. *Învățarea centrată pe elev. Ghid pentru profesori și formatori.* Proiectul PHARE: RO. IMC Consulting Ltd, 2005.
23. Manolescu, M., „*Evaluarea școlară. Metode, tehnici, instrumente*”, Editura Meteor Press, București, 2005.
24. Masalagiu Cristian, Asiminoaei Ioan, Țibu Mirela. *Didactica predării Informaticii.* Editura POLIROM, 2016.
25. Christiane Bosman, François-Marie Gérard, Xavier Roegiers. *Quel avenir pour les compétences?*, De Boesk & Larcier s.a., 2002.
26. Colis B. and Moonen J., *Flexible Learning in a Digital World: Experiences and expectations* (London: Kogan-Page), 2001.
27. Heather Fry, Steve Ketteridge, Stephanie Marshall, *A handbook for teaching and learning in education: enhancing academic practice*, Taylor & Francis, USA New York, 2009.
28. Malcolm Shepherd Knowles, *Self-directed learning: A guide for learners and teachers*, Englewood Cliffs: Prentice Hall/Cambridge, 1975.
29. Wilson, B. G. *Metaphors for instruction: Why we talk about learning environments.* Educational Technology, 35 (5), 25-30, (1995).