

РАБОЧАЯ ПРОГРАММА
учебного курса «Креативный код (информатика)»
для обучающихся 6-7 классов
(часть содержательного раздела ООП ООО)
Срок освоения 2 года

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Рабочая программа курса «Креативный код (информатика)» даёт представление о цели, задачах, общей стратегии обучения, воспитания и развития обучающихся средствами курса внеурочной деятельности по информатике, устанавливает содержание курса, предусматривает его структурирование по разделам и темам; предлагает распределение учебных часов по разделам и темам и последовательность их изучения с учётом межпредметных и внутрипредметных связей, логики учебного процесса и возрастных особенностей обучающихся, включает описание форм организации занятий и учебно-методического обеспечения образовательного процесса.

Обучение строится по принципу каскадного усложнения. В 6 классе акцент делается на изучении информационных процессов (двоичный код, единицы измерения) и отработке сложных алгоритмических конструкций (циклов с условием и параметром) в визуальной среде. В 7 классе накопленный визуальный опыт трансформируется в написание текстовых скриптов на языке Python. В программу 7 класса интегрирован специальный блок прикладных задач по алгебре и геометрии (алгоритм Евклида, НОД/НОК, логические связки, координатная графика), что делает код осязаемым, снимает страх перед программированием в ОГЭ и повышает общую ИТ-грамотность.

Курс «Креативный код (информатика)» отражает:

- сущность информатики как научной дисциплины, изучающей закономерности протекания и возможности автоматизации информационных процессов в различных системах;
- основные области применения информатики, прежде всего информационные технологии, управление и социальную сферу;
- междисциплинарный характер информатики и информационной деятельности.

Информатика характеризуется всё возрастающим числом междисциплинарных связей, причём как на уровне понятийного аппарата, так и на уровне инструментария. Современная школьная информатика оказывает существенное влияние на формирование мировоззрения школьника, его жизненную позицию, закладывает основы понимания принципов функционирования и использования информационных технологий как необходимого инструмента практически любой деятельности и одного из наиболее значимых технологических достижений современной цивилизации. Многие предметные знания и способы деятельности, освоенные обучающимися при изучении информатики, находят применение как в рамках образовательного процесса при изучении других предметных областей, так и в иных жизненных ситуациях, становятся значимыми для формирования качеств личности, т.е. ориентированы на формирование метапредметных и личностных результатов обучения.

Курс «Креативный код (информатика)» расширяет содержание четырёх тематических разделов информатики на уровне основного общего образования:

- цифровая грамотность;
- теоретические основы информатики;
- алгоритмы и программирование;
- информационные технологии.

Целями изучения курса являются:

- развитие алгоритмического и критического мышления, что предполагает способность обучающегося разбивать сложные задачи на более простые подзадачи;
- углубление алгоритмических компетенций учащихся, развитие навыков обработки структурированной информации, плавный переход к текстовому программированию на языке Python и целенаправленная пропедевтика ОГЭ по информатике.
- формирование цифровых навыков, в том числе ключевых компетенций цифровой экономики, таких как базовое программирование, основы работы с данными, коммуникация в современных цифровых средах, информационная безопасность; воспитание ответственного и избирательного отношения к информации;
- формирование необходимых для успешной жизни в меняющемся мире универсальных учебных действий (универсальных компетентностей) на основе средств и методов информатики и информационных технологий, в том числе овладение умениями работать с различными видами информации, самостоятельно планировать и осуществлять индивидуальную и коллективную информационную деятельность, представлять и оценивать её результаты;
- формирование и развитие компетенций обучающихся в области использования информационно-коммуникационных технологий, знаний, умений и навыков работы с информацией, программирования, коммуникации в современных цифровых средах в условиях обеспечения информационной безопасности личности обучающегося.

Основные задачи курса — сформировать у обучающихся:

- понимание принципов устройства и функционирования объектов цифрового окружения, представления об истории и тенденциях развития информатики периода цифровой трансформации современного общества;
- владение основами информационной безопасности;
- знания, умения и навыки грамотной постановки задач, возникающих в практической деятельности, их решение с помощью информационных технологий;
- умения и навыки формализованного описания поставленных задач;
- знание основных алгоритмических структур и умение применять эти знания для построения алгоритмов решения задач по их математическим моделям;
- умения и навыки эффективного использования основных типов прикладных программ (приложений) общего назначения и информационных систем для решения с их помощью практических задач;
- умение грамотно интерпретировать результаты решения практических задач с помощью информационных технологий, применять полученные результаты в практической деятельности.

МЕСТО КУРСА В УЧЕБНОМ ПЛАНЕ

Программа курса составлена из расчёта 34 часа:

6 класс- 0,5 часа 17 часов 7 класс – 0,5 часа 17 часов

2. СОДЕРЖАНИЕ ОБУЧЕНИЯ ПО ГОДАМ

6 КЛАСС (17 часов) — «Двоичный код, офисные технологии и сложные циклы»

Раздел 1. Компьютерная грамотность и безопасность (3 часа).

Типы и поколения ЭВМ (персональные, встроенные компьютеры, суперкомпьютеры). Иерархическая файловая система: полное имя файла, пути к каталогам, расширения файлов различных типов. Угрозы безопасности данных, вредоносное ПО, встроенные и сетевые антивирусные средства.

Раздел 2. Теоретические основы информатики (4 часа).

Информационные процессы: получение, хранение, поиск, обработка и передача информации. Двоичный код как основа представления данных в компьютере. Равномерный двоичный код, правила построения кодовых таблиц. Единицы измерения информации (от бита до гигабайта). Расчет информационного объема текстовых, графических и аудиофайлов (пропедевтика ОГЭ).

Раздел 3. Алгоритмизация и основы программирования (4 часа).

Углубленная работа в визуальной среде программирования. Реализация сложных циклических алгоритмов: циклы с предусловием. Циклы с параметром. Проектирование и отладка алгоритмов, совмещающих в себе одновременно вложенные циклы и ветвления.

Раздел 4. Информационные технологии и создание контента (5 часов).

Компьютерная графика: растровая и векторная. Создание векторных объектов встроенными средствами офисных пакетов. Текстовый редактор: работа со списками (маркированные, нумерованные). Добавление и форматирование таблиц в документах. Презентации: гиперссылки, интерактивные элементы управления, настройка анимации. Создание комплексного проекта «Круговая презентация».

Резерв и подведение итогов (1 час).

Обобщающее повторение, итоговая контрольная работа за 6 класс.

7 КЛАСС (17 часов) — «Текстовое программирование на Python и цифровое проектирование»

Раздел 1. Введение в текстовое программирование на Python (4 часа).

Интерфейс среды разработки. Алфавит и базовый синтаксис языка Python. Вывод данных на экран (print()). Понятие переменной в программировании (связь с переменной в алгебре). Ввод текстовых и числовых данных (input(), int()). Арифметические операции в Python: целочисленное деление (//) и остаток от деления (%). Проектирование линейных программ по математическим формулам.

Раздел 2. Условные алгоритмы и математическая логика (4 часа).

Реализация алгоритмов ветвления на языке Python. Операторы if, else, elif. Логические операции конъюнкции, дизъюнкции и отрицания (and, or, not). Применение логических связок для проверки математических гипотез. Синхронизированный практикум: написание программы для геометрической проверки неравенства треугольника.

Раздел 3. Циклы и прикладные математические алгоритмы (4 часа).

Реализация циклических конструкций в текстовом коде. Цикл с условием while. Интеграция с математикой: программирование алгоритма Евклида для автоматического нахождения наибольшего общего делителя (НОД). Нахождение наименьшего общего кратного (НОК) по математической формуле связи. Цикл с параметром for и функция перебора последовательностей range(). Написание скрипта для поиска простых чисел и вычисления факториала.

Раздел 4. Модульное программирование и цифровая разработка (4 часа).

Понятие вспомогательного алгоритма. Определение и вызов функций в Python (def), аргументы и возвращаемые значения (return). Знакомство с этапами ИТ-разработки (идея, ТЗ, проектирование логики). Разработка сценариев, интенты и условия в конструкторе чат-ботов Aimylogic. Применение инструментов искусственного интеллекта для оптимизации и отладки текстового кода.

Резерв и подведение итогов (1 час).

Публикация и защита мини-проектов (интеллектуальный математический чат-бот или консольная программа на Python).

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ КУРСА (ПО ГОДАМ)

Личностные результаты

У обучающегося будут сформированы:

- ответственное отношение к учению, готовность и способность к саморазвитию и самообразованию на основе мотивации к изучению информатики и программирования;
- устойчивый познавательный интерес к курсу как к средству развития алгоритмического и критического мышления;
- осознание значимости информационных технологий и цифровых компетенций для решения практических и жизненных задач;
- готовность к сотрудничеству и коммуникации, умение работать в команде при разработке проектов;
- креативность мышления, инициативность, находчивость и активность при поиске способов решения задач с помощью кода;
- способность к самооценке и рефлексии собственной учебной деятельности, умение принимать право на ошибку и исправлять её;
- готовность применять цифровые навыки для решения задач, связанных с повседневной жизнью и будущей профессиональной деятельностью.

Метапредметные результаты

Познавательные универсальные учебные действия:

- выявлять дефицит информации, необходимой для решения поставленной задачи;
- преобразовывать данные из схем, таблиц и диаграмм в текстовый формат и обратно;
- самостоятельно строить алгоритмы решения расчётных и логических задач;
- анализировать условие задачи, выделять известные и неизвестные данные, устанавливать связи между ними;
- переводить текстовую информацию в схему, таблицу, математическую модель, блок-схему алгоритма;

- использовать сравнение, аналогию, перебор вариантов, классификацию, обобщение и доказательное рассуждение при решении задач;
- выбирать способ решения из нескольких возможных, соотнося его с условием задачи и собственными возможностями;
- интерпретировать результаты решения, соотносить их с условием и оценивать их разумность;
- использовать различные источники информации, представлять и преобразовывать информацию в удобную для решения задачи форму.
- Коммуникативные универсальные учебные действия:
- ясно, точно и грамотно формулировать свои мысли в устной и письменной форме при объяснении хода решения;
- задавать вопросы по существу обсуждаемой задачи, высказывать предположения и предложения;
- участвовать в совместном поиске решения, обсуждать различные способы, сопоставлять и аргументированно оценивать их;
- договариваться о распределении ролей в парной и групповой работе, планировать совместную деятельность;
- представлять результаты решения задачи, кратко и логично комментировать полученный ответ;
- использовать математическую и алгоритмическую терминологию и символику в учебном диалоге и при защите решения.
- Регулятивные универсальные учебные действия:
- ставить цель решения задачи и понимать, какой результат необходимо получить;
- самостоятельно планировать последовательность действий при решении задачи;
- выбирать рациональный способ решения с учётом условий задачи и имеющихся ресурсов;
- осуществлять самоконтроль на всех этапах решения, проверять вычисления и логику рассуждений;
- вносить коррективы в решение при обнаружении ошибки или изменении условий;
- оценивать правильность и полноту выполнения задания;
- прогнозировать результат и проверять его соответствие условию;
- проявлять настойчивость при столкновении с затруднением, искать способ преодоления трудности (в том числе при отладке программного кода);
- владеть навыками дебаггинга (поиск и исправление ошибок в алгоритмах на основе несоответствия результата и цели);
- адаптировать решение к меняющимся условиям.

ПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ

6 КЛАСС

По окончании изучения курса в 6 классе обучающийся научится:

- ориентироваться в иерархической файловой системе компьютера, выполнять операции копирования, перемещения, удаления и поиска файлов по расширениям;
- применять базовые меры по защите персональных данных от вредоносного ПО;
- свободно оперировать единицами измерения информации, переводить значения из мелких в более крупные и обратно;
- преобразовывать текстовую и графическую информацию в формат двоичного кода;
- рассчитывать информационный объём сообщения (текста, изображения) по формулам;

- составлять, выполнять вручную и на компьютере циклические алгоритмы (с условием и с параметром) для управления исполнителями в визуальной среде;
- различать растровую и векторную графику, создавать графические объекты по собственному замыслу;
- структурировать текст с помощью списков, добавлять и форматировать таблицы в текстовом процессоре;
- создавать интерактивные презентации с настроенной анимацией и рабочими гиперссылками.
- По окончании изучения курса в 6 классе обучающийся получит возможность научиться:
- применять полученные знания для создания комплексных мультимедийных проектов;
- самостоятельно отлаживать алгоритмы в визуальной среде программирования;
- использовать приобретённые навыки для решения задач из других предметных областей.

7 КЛАСС

По окончании изучения курса в 7 классе обучающийся научится:

- писать синтаксически корректный код на языке Python с использованием переменных различных типов данных (строки, целые числа);
- применять операторы `input()` и `print()`, производить явное приведение типов данных (`int()`);
- использовать операции целочисленного деления (`//`) и остатка от деления (`%`) для решения арифметических задач;
- реализовывать ветвления с помощью операторов `if`, `else`, `elif`;
- составлять сложные условия с использованием логического умножения, сложения и отрицания (`and`, `or`, `not`);
- программировать циклы `while` и `for` (вместе с функцией `range()`);
- писать на Python программы автоматического расчёта НОД (алгоритм Евклида), НОК, факториала и проверки чисел на простоту;
- управлять исполнителем Черепашка (`turtle`) по координатам на плоскости, строить графики линейных функций ($y=kx+b$) по точкам;
- объявлять и вызывать вспомогательные алгоритмы (функции) с параметрами и возвращаемым значением (`return`);
- проектировать разветвлённые диалоговые сценарии чат-ботов в среде `Aimylogic` с применением переменных;
- использовать возможности нейросетей для проверки логики кода.
- По окончании изучения курса в 7 классе обучающийся получит возможность научиться:
- самостоятельно проектировать и реализовывать программные продукты (консольные утилиты, чат-боты);
- применять полученные алгоритмические навыки для решения задач повышенного уровня сложности;
- оценивать эффективность выбранного способа решения и сравнивать разные стратегии программирования;
- работать в группе над ИТ-проектом, обосновывать свою позицию и учитывать мнение других участников обсуждения.

4. ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ КУРСА

6 КЛАСС

Тема	Содержание программы
Компьютерная грамотность и безопасность	Классификация и типы компьютеров (персональные, встроенные, суперкомпьютеры). Продвинутая работа с иерархической файловой системой, ветвление каталогов, полные пути к файлам. Угрозы безопасности устройства, защита от вредоносного программного обеспечения, штатные и сторонние антивирусные средства.
Информационные процессы	Подробный разбор и классификация информационных процессов: сбор (получение), хранение, обработка и передача информации в живой природе, обществе и технике.
Двоичный код	Знакомство с двоичным кодом. Представление различных видов данных в компьютере в виде универсального двоичного алфавита. Равномерный двоичный код.
Единицы измерения информации	Единицы измерения информации (бит, байт, килобайт, мегабайт, гигабайт). Связь между единицами. Информационный объём текстовых и графических данных. Характерные размеры файлов различных типов. Решение базовых задач по расчету объема данных (пропедевтика ОГЭ).
Алгоритмизация и основы программирования	Реализация циклических алгоритмов в среде визуального блочного программирования. Циклы с условием (пока условие истинно). Циклы с параметром (повтори N раз). Решение комплексных логических задач на совместное одновременное использование циклов и ветвлений (пропедевтика программирования в ОГЭ).
Компьютерная графика	Виды компьютерной графики: растровая и векторная графика, их ключевые различия и особенности масштабирования. Практика создания векторных рисунков из геометрических примитивов.
Текстовый процессор	Продвинутая работа в текстовом процессоре: структурирование сложной информации с помощью маркированных и нумерованных списков. Добавление, редактирование и форматирование таблиц в текстовых документах. Комплексное оформление текстовых документов.
Мультимедиа и презентации	Презентации: детальная настройка анимации объектов на слайде, интерактивные элементы управления и использование внутренних/внешних гиперссылок. Проектная работа «Круговая презентация».
Резерв и подведение итогов	Обобщающее повторение ключевых тем курса, систематизация знаний. Выполнение итоговой контрольной работы за курс 6 класса.

7 КЛАСС

Тема	Содержание программы
Среда разработки и базовый синтаксис	Понимание различий между визуальной (блочной) и текстовой средами программирования. Интерфейс интегрированной среды разработки (IDE). Алфавит, базовый синтаксис и правила оформления читаемого кода на языке Python. Комментарии в коде.

Операции ввода и вывода данных	Использование операций вывода данных на экран (print()). Понятие переменной в программировании, ее имя и значение (связь с понятием переменной в алгебре). Операции ввода текстовых и числовых данных с клавиатуры (input()). Явное приведение типов данных к целочисленному (int()).
Арифметические операции в коде	Применение в коде стандартных арифметических операций. Специальные операции: целочисленное деление (//) и взятие остатка от деления (%). Применение этих операций для анализа математических свойств чисел (проверка чётности, выделение отдельных разрядов и цифр в числе).
Практикум линейного программирования	Проектирование и написание линейных программ по готовым математическим формулам. Вычисление периметра, площади и объёма по входным данным. Пропедевтика математических формул за курс 7 класса.
Конструкции ветвления	Реализация алгоритмов ветвления на языке Python. Синтаксис и правила отступов для операторов if (если) и else (иначе). Множественное ветвление с использованием оператора elif.
Операторы математической логики	Изучение основ математической логики. Составные логические выражения. Использование в условных операторах логических связок конъюнкции (and), дизъюнкции (or) и отрицания (not).
Синхронизированный практикум	Применение логических связок для автоматической проверки математических гипотез. Написание программы для геометрической проверки неравенства треугольника (существует ли треугольник с заданными сторонами).
Цикл с условным перебором	Реализация циклических конструкций в текстовом коде. Синтаксис и особенности работы цикла с условием while. Защита от бесконечного закликивания кода.
Интеграция с математикой: алгоритмы	Интеграция программирования с курсом математики. Написание скрипта для реализации алгоритма Евклида (автоматическое нахождение наибольшего общего делителя — НОД). Вычисление наименьшего общего кратного (НОК) по математической формуле связи.
Цикл с параметром и перебор последовательностей	Изучение цикла с параметром for и встроенной функции генерации последовательностей чисел range(). Особенности указания старта, стопа и шага перебора.
Скрипты для анализа чисел	Написание программного кода для автоматического поиска простых чисел в заданном диапазоне методом перебора делителей. Написание скрипта для вычисления факториала числа.
Вспомогательные алгоритмы и графический модуль	Понятие вспомогательного алгоритма в текстовом программировании. Определение и вызов собственных функций в Python с помощью оператора def. Передача аргументов в функцию и использование возвращаемых значений (return). Программирование траекторий движения графического исполнителя (модуль turtle / Черепашка) по координатной плоскости. Автоматизация построения графиков линейных функций ($y = kx + b$).
Проектирование ИТ-продуктов и чат-боты	Знакомство с этапами реальной ИТ-разработки (идея, составление технического задания, проектирование логики, тестирование). Разработка линейных и разветвляющихся сценариев диалогов (интенты, экраны, переменные) в конструкторе чат-ботов Aimylogic.
Искусственный интеллект в веб-разработке	Применение доступных инструментов искусственного интеллекта (нейросетей) для анализа написанного кода, дебаггинга (поиска синтаксических и логических ошибок) и оптимизации алгоритмов.

Проектный модуль и резерв времени	Публикация, демонстрация и защита готовых мини-проектов (интеллектуальный математический чат-бот в Aimylogic или консольная программа/игра на Python). Систематизация знаний.
-----------------------------------	---

КТП Информатика 6 класс (17 часов)

Номер урока	Тема урока
1	Типы компьютеров. Иерархическая файловая система и работа с файлами
2	Основы безопасности компьютера и безопасная работа в интернете
3	Информационные процессы.
4	Преобразование информации. Знакомство с двоичным кодом
5	Преобразования с двоичным кодом
6	Информационный объём данных и единицы измерения информации
7	Циклические алгоритмы: реализация цикла с условием в среде программирования
8	Циклические алгоритмы: реализация цикла с параметром в среде программирования
9	Решение сложных задач на сочетание циклов и ветвлений
10	Переменные и их программирование
11	Виды компьютерной графики: растровая и векторная графика
12	Основы работы с текстовыми документами и структурирование текстов с помощью списков
13	Добавление таблиц и комплексная работа со списками, таблицами и картинками в документах
14	Компьютерные презентации: анимация, интерактивные элементы и гиперссылки
15	Разработка презентаций
16	Проект «Круговая презентация»
17	Обобщающее повторение материала и Итоговая контрольная работа за 6 класс

КТП Информатика 7 класс (17 часов)

Номер урока	Тема урока
1	Различия между визуальной и текстовой средами программирования. Интерфейс сред разработки . Базовый синтаксис.
2	Переменные в программировании и их связь с переменными в алгебре. Операции ввода и вывода данных (print(), input()).
3	Типы данных. Приведение к целочисленному типу (int()). Специальные арифметические операции: целочисленное деление (//) и остаток (%).

4	Практикум линейного программирования: написание программ расчёта периметра, площади и объёма по математическим формулам.
5	Реализация алгоритмов ветвления в текстовом коде. Конструкция if - else. Синтаксис и правила отступов в Python.
6	Множественное ветвление с использованием оператора elif. Примеры каскадных проверок.
7	Основы математической логики в коде. Логические связки конъюнкции, дизъюнкции и отрицания (and, or, not).
8	Синхронизированный практикум: написание программы для геометрической проверки гипотезы о неравенстве треугольника.
9	Реализация циклических конструкций. Цикл с условием while. Защита от бесконечного заикливания кода.
10	Интеграция с математикой: программирование алгоритма Евклида для поиска НОД. Вычисление НОК по формуле связи.
11	Цикл с параметром for и встроенная функция перебора последовательностей range(). Параметры: старт, стоп, шаг.
12	Математический скриптинг: написание кода для поиска простых чисел методом перебора делителей и вычисления факториала.
13	Понятие функции (def). Аргументы функции и возвращаемые значения (return). Программирование траекторий графического исполнителя (модуль turtle / Черепашка).
14	Автоматизация построения графиков линейных функций ($y = kx + b$) средствами программного кода Python.
15	Проектирование ИТ-продуктов: этапы разработки от ТЗ до отладки. Разработка сценариев диалогов и условий в конструкторе чат-ботов Aimylogic.
16	Применение инструментов искусственного интеллекта (ИИ-ассистентов) для дебаггинга кода, поиска синтаксических ошибок и оптимизации.
17	Публикация и защита мини-проектов (интеллектуальный математический чат-бот или консольная утилита). Итоговое тестирование за курс.