

Муниципальное общеобразовательное учреждение
«Средняя школа № 86 Тракторозаводского района Волгограда»

Выписка из основной образовательной программы среднего общего образования

РАБОЧАЯ ПРОГРАММА УЧЕБНОГО КУРСА
«Избранные вопросы по информатике»
для 11 классов

Выписка верна 30.08.2024

Директор



Е.П. Дьячкова

Пояснительная записка.

Предлагаемая программа учебного курса «Избранные вопросы по информатике» предназначена для обучающихся 11 классов средней школы.

При разработке программы данного курса учитывалось то, что она как компонент образования должна быть направлена на удовлетворение потребностей старшеклассников, развитие их познавательной активности, потенциальных творческих способностей и повышение интереса к информатике, на формирование у обучающихся новых видов практической деятельности, которые не характерны для традиционных учебных курсов.

Данный учебный курс дополняет программу учебного предмета «Информатика» на базовом уровне и предназначен для учащихся, которые проявляют интерес к изучению программирования, т.к. на базовом уровне этот раздел информатики представлен не достаточно полно.

Раздел «Алгоритмизация и программирование» входит в обязательный минимум содержания основной образовательной программы по информатике. Вопросы, рассматриваемые в данном курсе, выходят за рамки обязательного содержания. Вместе с тем, они тесно примыкают к основной программе по информатике. Поэтому данный курс будет способствовать совершенствованию и развитию важнейших знаний и умений в области информатики, предусмотренных школьной программой, поможет оценить свои возможности по информатике и более осознанно выбрать профиль дальнейшего обучения.

Курс рассчитан на базовый уровень владения обучающимися знаниями в области алгоритмизации и программирования. Он является предметно-ориентированным, расширяющим темы базовых образовательных программ.

Класс: 11.

Количество часов в неделю: 2 ч в неделю, всего 68 учебных часов.

Образовательная область: «Информатика»

Изучение программирования связано с развитием целого ряда таких умений и навыков, которые носят обще-интеллектуальный характер и формирование которых – одна из приоритетных задач современной школы. Изучение программирования развивает мышление школьников, способствует формированию у них многих приемов умственной деятельности. Здесь роль информатики сродни роли математики в школьном образовании. Изучая программирование, учащиеся прочнее усваивают основы алгоритмизации, приобщаются к алгоритмической культуре, познают азы профессии программиста.

Тематика данного учебного курса предопределяет превалирование в его структуре практических и проектных занятий.

Цель: - формирование у обучающихся целостного представления об алгоритмизации и программировании, системы знаний, умений и навыков решения задач повышенной сложности на языке программирования Python.

Задачи:

- Предоставить учащимся возможность проявить интерес к изучению программирования для дальнейшего применения в различных профессиональных областях.
- Предоставить учащимся возможность систематизировать представление о структурном и объектно-ориентированном программировании.
- Познакомить учащихся со всевозможными методами решения задач, реализуемых на языке программирования Python.
- Продолжить работу по развитию логического и алгоритмического мышления старшеклассников.
- Продолжить формирование информационной культуры обучающихся, навыков грамотной разработки программ.

- Продолжить работу по формированию у обучающихся умений и навыков решения задач по программированию и алгоритмизации повышенной сложности.

Программа обеспечивает достижение следующих результатов освоения:

Личностные:

- сформированность мировоззрения, соответствующего современному уровню развития науки и техники;
- готовность и способность к образованию, в том числе самообразованию, на протяжении всей жизни; сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности;
- понимание роли программирования в современном мире; способность связать учебное содержание с собственным жизненным опытом, понять значимость подготовки в области программирования в условиях развития информационного общества;
- осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов;
- отношение к профессиональной деятельности как возможности участия в решении личных, общественных, государственных, общенациональных проблем.

Метапредметные результаты:

- умение самостоятельно определять цели деятельности и составлять планы деятельности; самостоятельно осуществлять, контролировать и корректировать деятельность; использовать все возможные ресурсы для достижения поставленных целей и реализации планов деятельности; выбирать успешные стратегии в различных ситуациях;
- умение продуктивно общаться и взаимодействовать в процессе совместной деятельности, учитывать позиции других участников деятельности, эффективно разрешать конфликты;
- владение навыками познавательной, учебно-исследовательской и проектной деятельности, навыками разрешения проблем; способность и готовность к самостоятельному поиску методов решения практических задач, применению различных методов познания;
- готовность и способность к самостоятельной информационно-познавательной деятельности, включая умение ориентироваться в различных источниках информации, критически оценивать и интерпретировать информацию, получаемую из различных источников;
- умение использовать средства информационных и коммуникационных технологий в решении когнитивных, коммуникативных и организационных задач с соблюдением требований эргономики, техники безопасности, гигиены, ресурсосбережения, правовых и этических норм, норм информационной безопасности.

Предметные результаты:

Обучающийся научится:

- определять основные алгоритмические конструкции и правила их записи, основные способы организации данных;
- использовать стандартные алгоритмические конструкции при программировании;
- формально исполнять алгоритмы, записанные на естественных и алгоритмических языках, в том числе на языках программирования;
- составлять и записывать алгоритмы с использованием соответствующих алгоритмических конструкций;
- анализировать обстановку исполнителя алгоритма;
- анализировать результат исполнения алгоритма;
- распознавать необходимость применения той или иной алгоритмической конструкции при решении задачи;
- организовывать данные для эффективной алгоритмической обработки;
- разрабатывать алгоритмы и реализовывать их на языке программирования Python;
- осуществлять отладку и тестирование программы.

Обучающийся получит возможность научиться:

- создавать алгоритм решения задач уровня выше среднего и высокого;
- анализировать текст программы с точки зрения соответствия записанного алгоритма поставленной задаче и изменять его в соответствии с заданием;
- реализовывать сложный алгоритм с использованием современных систем программирования.

Содержание учебного курса

Содержание учебного курса «Избранные вопросы по информатике» включает следующие разделы:

1. Основы программирования на языке Python – 12 часов
2. Графический модуль Turtle в языке программирования Python – 6 часов
3. Функции и события на примере модуля Turtle в языке программирования Python – 8 часов
4. Работа с файлами – 2 часа
5. Решение задач повышенной сложности на использование различных алгоритмических конструкций, подпрограмм на языке Python – 7 часов
6. Массивы данных – 13 часов
7. Работа со строками и множествами – 8 часов
8. Объектно-ориентированное программирование на примере языка Python – 8 ч
9. Решение сложных задач по программированию – 4 часа

Раздел 1.

Основы программирования на языке Python: структура программы, алфавит, правила записи имен. Меню, некоторые операции с текстом, запуск программы. Оператор присваивания. Переменные, типы данных. Арифметические выражения. Операторы ввода и вывода данных. Стандартные функции. Операции с целыми числами. Условный оператор. Составные операторы. Оператор выбора. Программирование циклических алгоритмов (цикл с предусловием, цикл с постусловием, цикл с параметром). Тело цикла. Подпрограммы (процедуры и функции), их описание и вызов в программе.

Раздел 2.

Подключение модуля Turtle. Объект. Метод. Основные команды управления черепашкой. Заливка замкнутых многоугольников. Рисование окружности. Изменение внешности черепашки при помощи команды Shape. Управление несколькими черепашками.

Раздел 3.

Функция, виды функций. Функции модуля Turtle. Самостоятельное создание функции. Глобальные и локальные переменные. Объект «экран». Событие. Работа с событиями. Фракталы. Рекурсия. Кривая Коха

Раздел 4.

Работа с файлами. Открытие, закрытие файла. Чтение и запись файлов в Python. Расширенная работа с файлами в Python.

Раздел 5.

Задачи повышенной сложности с математическим содержанием на использование различных алгоритмических конструкций, использование подпрограмм. Прямая рекурсия, взаимная рекурсия, примеры задач на рекурсию, не рекурсивные способы (использование циклов, стека и т. д.) решения рекурсивных задач.

Раздел 6.

Массивы (одномерные (линейные) и двумерные), различные способы их описания в программе. Обработка массивов (ввод и вывод элементов массива; поиск элементов в массиве; проведение математических операций с элементами массива; замена, удаление и вставка элементов в массиве; сортировка; реверс, циклический сдвиг, срезы). Решение задач

повышенной сложности на обработку массивов, двоичный поиск. Матрицы, задачи обработки матриц. Квадратные матрицы, перестановка строк и столбцов, выделение строк и столбцов.

Раздел 7.

Символьные строки. Операции со строками (объединение, срезы, срезы с отрицательными индексами, удаление и вставка символов, стандартные функции, поиск в строках, преобразования «строка» - «число»). Строки в процедурах и функциях. Замена подстроки, замена всех экземпляров подстроки. Рекурсивный перебор в строках. Сортировка строк.

Раздел 8.

Основные понятия объектно-ориентированного программирования. Классы в Python, атрибуты классов. Объектно-ориентированный интерфейс. Наследование, полиморфизм, Инкапсуляция в Python. Модули в Python.

Раздел 9.

Решение сложных задач по программированию.

Основными методами обучения являются практические методы при выполнении заданий практикума и метод проектов, позволяющий синтезировать полученные знания по программированию на языке Python и применению их при решении. Практические работы позволяют отработать основные умения по программированию. Роль учителя состоит в кратком по времени объяснении нового материала и постановке задачи, а затем консультировании учащихся в процессе выполнения практического задания.

Контроль знаний и умений. Текущий контроль уровня усвоения материала осуществляется по результатам выполнения учащимися заданий практикума. Уровень достижения образовательных результатов фиксируется через рейтинговую систему (показателем которой является сумма баллов, полученных школьником за выполнение практических работ). Итоговый контроль реализуется в форме выполнения проекта по созданию собственного приложения на языке программирования Python.

Организация учебного процесса.

Предусматривается организация учебного процесса в двух взаимосвязанных и взаимодополняющих формах:

- урочная форма, в которой учитель объясняет новый материал и консультирует учащихся в процессе выполнения ими практических заданий на компьютере;
- внеурочная форма, в которой учащиеся после уроков (дома или в школьном компьютерном классе) самостоятельно выполняют на компьютере практическое задание.

Учебно-тематическое планирование.

№ п/п	Название темы (количество часов)	Форма проведения	Содержание	Форма контроля	Дата проведе ния
Раздел 1. Основы программирования на языке Python (12 часов)					
1	Введение в язык программирования Python.	Лекция	Структура программы, алфавит, правила записи имен. Меню, некоторые операции с текстом, запуск программы. Операторы ввода и вывода данных.	Беседа	
2	Переменные в языке программирования Python.	Лекция. Практикум	Оператор присваивания. Переменные, типы данных.	Контроль выполнения заданий практикума	
3	Арифметические выражения	Лекция. Практикум	Арифметические выражения	Контроль выполнения заданий практикума	
4	Вычисления	Лекция. Практикум	Остаток от деления, целая часть от деления, вещественные числа	Контроль выполнения заданий практикума	
5	Стандартные функции	Лекция. Практикум	Стандартные функции	Контроль выполнения заданий практикума	
6	Математические функции	Лекция. Практикум	Математические функции, случайные числа	Контроль выполнения заданий практикума	
7	Разветвляющиеся алгоритмы на языке программирования Python. Условный оператор	Лекция. Практикум	Условный оператор. Полное и неполное ветвление.	Беседа по теоретической части	
8	Разветвляющиеся алгоритмы на языке программирования Python. Составной оператор	Лекция. Практикум	Составные операторы	Контроль выполнения заданий практикума	
9	Разветвляющиеся алгоритмы на языке программирования Python. Оператор выбора.	Лекция. Практикум	Оператор выбора.	Контроль выполнения заданий практикума	
10	Циклические алгоритмы на языке	Лекция. Практикум	Программирование циклических алгоритмов	Контроль выполнения	

	программирования Python. Цикл с предусловием		(цикл с предусловием). Тело цикла.	заданий практикума	
11	Циклические алгоритмы на языке программирования Python. Цикл с постусловием	Лекция. Практикум	Программирование циклических алгоритмов (цикл с постусловием).	Контроль выполнения заданий практикума	
12	Циклические алгоритмы на языке программирования Python. Цикл с параметром	Лекция. Практикум	Программирование циклических алгоритмов (цикл с параметром).	Контроль выполнения заданий практикума	
Раздел 2. Графический модуль Turtle в языке программирования Python (6 часов)					
13	Графические примитивы	Лекция	Графические примитивы. Знакомство с библиотекой Turtle. Объект. Метод. Основные команды управления черепашкой	Беседа по теоретической части	
14	Отрисовка графических примитивов	Лекция. Практикум	Заливка замкнутых многоугольников. Рисование окружности. Изменение внешности черепашки при помощи команды Shape.	Контроль выполнения заданий практикума	
15	Отличие GUI (Graphical User Interface) от графических примитивов.	Лекция	GUI (Graphical User Interface). Элементы GUI (кнопки, поля, метки и т.д.)	Беседа по теоретической части	
16	Решение задач с использованием графического модуля Turtle	Практикум	Практика условных операторов с помощью Turtle.	Контроль выполнения заданий практикума	
17	Решение задач с использованием графического модуля Turtle	Практикум	Практика циклических операторов с помощью Turtle.	Контроль выполнения заданий практикума	
18	Решение задач с использованием графического модуля Turtle	Практикум	Управление несколькими черепашками.	Контроль выполнения заданий практикума	
Раздел 3. Функции и события на примере модуля Turtle в языке программирования Python (8 часов)					
19	Функция в Python	Лекция	Функция, виды функций	Беседа по теоретической части	
20	Функции модуля Turtle	Лекция	Функции модуля Turtle	Беседа по теоретической части	
21	Создание функции.	Лекция.	Глобальные и локальные	Контроль	

		Практикум	переменные. Объект «экран».	выполнения заданий практикума	
22	Создание функции	Лекция. Практикум	Событие. Работа с событиями.	Контроль выполнения заданий практикума	
23	Создание функции	Лекция. Практикум	Самостоятельное создание функции	Контроль выполнения заданий практикума	
24	Фракталы.	Лекция. Практикум	Фракталы (понятие, применение)	Контроль выполнения заданий практикума	
25	Рекурсия.	Лекция. Практикум	Рекурсия (понятие, использование в задачах)	Контроль выполнения заданий практикума	
26	Снежинка Коха	Лекция. Практикум	Кривая Коха	Контроль выполнения заданий практикума	
Раздел 4. Работа с файлами (2 часа)					
27	Работа с файлами	Лекция. Практикум	Работа с файлами. Открытие, закрытие файла. Чтение и запись файлов в Python	Контроль выполнения заданий практикума	
28	Расширенная работа с файлами в Python	Лекция. Практикум	Расширенная работа с файлами в Python	Контроль выполнения заданий практикума	
Раздел 5. Решение задач повышенной сложности на использование различных алгоритмических конструкций, подпрограмм (7 часов)					
29	Решение задач повышенной сложности на использование условного оператора в Python	Лекция. Практикум	Задачи повышенной сложности с математическим содержанием на использование условного оператора в Python	Контроль выполнения заданий практикума	
30	Решение задач повышенной сложности на использование составных условий в Python	Практикум	Задачи повышенной сложности с математическим содержанием на использование составных условий в Python	Контроль выполнения заданий практикума	
31	Решение задач повышенной сложности на использование оператора выбора в Python	Практикум	Задачи повышенной сложности с математическим содержанием на использование оператора выбора в Python	Контроль выполнения заданий практикума	
32	Решение задач	Практикум	Задачи повышенной	Контроль	

	повышенной сложности на использование оператора цикла с предусловием и постусловием в Python		сложности с математическим содержанием на использование оператора цикла с предусловием и постусловием в Python	выполнения заданий практикума	
33	Решение задач повышенной сложности на использование оператора цикла с параметром в Python	Практикум	Задачи повышенной сложности с математическим содержанием на использование оператора цикла с параметром в Python	Контроль выполнения заданий практикума	
34	Решение задач на рекурсию.	Лекция. Практикум	Прямая рекурсия, взаимная рекурсия, примеры задач на рекурсию	Контроль выполнения заданий практикума	
35	Решение задач на рекурсию.	Лекция. Практикум	не рекурсивные способы (использование циклов, стека и т. д.) решения рекурсивных задач	Контроль выполнения заданий практикума	
Раздел 6. Массивы данных (13 часов)					
36	Массивы	Лекция. Практикум	Массивы, массивы в Python: списки, генераторы списков. Добавление элементов в список, удаление элементов из списка. Ввод массива с клавиатуры, вывод массива на экран.	Контроль выполнения заданий практикума	
37	Обработка одномерного массива	Лекция. Практикум	Обработка массивов (заполнение массива случайными числами и вывод элементов массива; перебор элементов)	Контроль выполнения заданий практикума	
38	Обработка одномерного массива	Лекция. Практикум	Обработка массивов (подсчёт количества нужных элементов, сумма элементов массива)	Контроль выполнения заданий практикума	
39	Алгоритмы обработки массивов	Лекция. Практикум	Обработка массивов (поиск в массиве; проведение математических операций с элементами массива, максимальный элемент и его номер).	Контроль выполнения заданий практикума	
40	Алгоритмы обработки массивов: реверс	Лекция. Практикум	Реверс массива	Контроль выполнения заданий	

				практикума	
41	Алгоритмы обработки массивов: циклический сдвиг	Лекция. Практикум	Циклический сдвиг элементов массива	Контроль выполнения заданий практикума	
42	Алгоритмы обработки массивов: срезы	Лекция. Практикум	Срезы в массиве	Контроль выполнения заданий практикума	
43	Двумерные массивы. Сортировка	Лекция. Практикум	Двумерные массивы: описание и ввод элементов, действия над ними. Поиск, замена в массиве. Сортировка массива. Способы сортировки.	Контроль выполнения заданий практикума	
44	Решение задач повышенной сложности на массивы.	Практикум	Задачи повышенной сложности на обработку массивов. Двоичный поиск. Особенности работы со списками	Контроль выполнения заданий практикума	
45	Матрицы	Лекция. Практикум	Матрицы, создание матриц, ввод матрицы с клавиатуры, вывод матриц	Контроль выполнения заданий практикума	
46	Поиск в матрице элемента, удовлетворяющего условию	Лекция. Практикум	Поиск в матрице элемента, удовлетворяющего условию	Контроль выполнения заданий практикума	
47	Поиск в матрице максимального элемента	Лекция. Практикум	Поиск в матрице максимального элемента	Контроль выполнения заданий практикума	
48	Квадратные матрицы.	Лекция. Практикум	Квадратные матрицы. Перестановка строк и столбцов, выделение строк и столбцов.	Контроль выполнения заданий практикума	
Раздел 7. Работа со строками и множествами (8 часов)					
49	Символьные строки	Лекция.	Символьные строки.		
50	Операции со строками	Лекция. Практикум	Операции со строками (объединение, срезы, срезы с отрицательными индексами, удаление и вставка символов)	Контроль выполнения заданий практикума	
51	Операции со строками	Лекция. Практикум	Операции со строками (стандартные функции, поиск в строках)	Контроль выполнения заданий практикума	
52	Операции со строками	Лекция. Практикум	преобразование «строка»-«число»	Контроль выполнения заданий практикума	

53	Строки в процедурах и функциях	Лекция. Практикум	Строки в процедурах и функциях	Контроль выполнения заданий практикума	
54	Замена подстроки	Лекция. Практикум	Замена подстроки, замена всех экземпляров подстроки	Контроль выполнения заданий практикума	
55	Рекурсивный перебор	Лекция. Практикум	Рекурсивный перебор в строках	Контроль выполнения заданий практикума	
56	Сортировка строк	Лекция. Практикум	Сортировка строк	Контроль выполнения заданий практикума	
Раздел 8. Объектно-ориентированное программирование на примере языка Python (8 часов)					
57	Основные понятия объектно-ориентированного программирования	Лекция	Основные понятия объектно-ориентированного программирования. Классы, атрибуты классов.	Беседа по теоретической части	
58	Объектно-ориентированный интерфейс	Лекция	Объектно-ориентированный интерфейс	Беседа по теоретической части	
59	Наследование в Python	Лекция. Практикум	Наследование (понятие, применение)	Контроль выполнения заданий практикума	
60	Полиморфизм в Python	Лекция. Практикум	Полиморфизм параметров, полиморфизм операторов, полиморфизм наследования, полиморфизм функций	Контроль выполнения заданий практикума	
61	Использование наследования для достижения полиморфизма	Лекция. Практикум	Виртуальный метод, метод перезагрузки операторов, магический метод	Контроль выполнения заданий практикума	
62	Инкапсуляция в Python	Лекция. Практикум	Инкапсуляция (понятие, применение)	Контроль выполнения заданий практикума	
63	Модули в Python	Лекция	Модули		
64	Создание собственного модуля в Python	Практикум	Создание собственного модуля	Контроль выполнения заданий практикума	
9. Решение сложных задач по программированию (4 часа)					
65-	Решение сложных	Практикум	Сложные задачи по	Контроль	

66	задач по программированию		программированию	выполнения заданий практикума	
67-68	Создание собственного приложения на языке Python	Проект	Проект «Создание собственного приложения на языке Python»	Представление проекта	

Список используемых ресурсов:

1. Бизли Д., Джонс Б. К. Python. Книга рецептов / пер. с англ. Б. В. Уварова. – М.: ДМК Пресс, 2019. – 648 с. 50
2. Доусон М. Програмируем на Python, 3-е издание – Пер. с англ. – СПб.: Питер, 2014. – 650 с.
3. Зингаро Д. Python без проблем: решаем реальные задачи и пишем полезный код. Издательство: Питер, 2023. – 336 с.
4. Лутц М. Изучаем Python, 3-е издание – Пер. с англ. – СПб.: Символ-Плюс, 2009. – 848 с.
5. Николенко С. И., Кадулин А. А., Архангельская Е. О. Глубокое обучение: погружение в мир нейронных сетей. Издательство: Питер, 2022. – 480 с.
6. Персиваль Г. Python. Разработка на основе тестирования / Г. Персиваль. – Москва: Издательство ДМК Пресс, 2018. – 622с.
7. Рашид, Тарик. Создаем нейронную сеть.: Пер. с англ. — СПб.: ООО «Диалектика», 2019. — 272 с.
8. Чернышев, С. А. Основы программирования на Python: учебное пособие для среднего профессионального образования / С. А. Чернышев. — Москва: Издательство Юрайт, 2023. — 286 с.

Электронные информационные ресурсы:

1. <https://pythonworld.ru/samouchitel-python> (сайт по изучению основ программирования на Python);
2. <https://pypi.org/search/?q=Pyqt> (сайт с кратким описанием библиотек).

Электронные образовательные ресурсы:

1. <https://bosova.ru/metodist/authors/informatika/3/python.php> (Python: графика и анимация)
2. <https://kpolyakov.spb.ru/school/pycpp.htm> (К. Поляков. Программирование. Python, C++)
3. <https://www.python.org/downloads/> (интерпретатор Python);
4. <https://www.jetbrains.com/pycharm/download/#section=windows> (среда разработки Pycharm);