

Пояснительная записка

Программа внеурочной деятельности по информатике составлена в соответствии с:

1. Федеральным Законом от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;
 2. Федеральным государственным образовательным стандартом среднего общего образования, утвержденным приказом Министерства образования и науки РФ от 17 мая 2012 г. N 413 с изменениями и дополнениями от: 29 декабря 2014 г., 31 декабря 2015 г., 29 июня 2017 г., 24 сентября, 11 декабря 2020 г.
 3. Основной образовательной программой среднего общего образования МБОУ СОШ № 72
- Раздел «Программирование» в школьном курсе представлен языком программирования Pascal, а многим обучающимся хочется познакомиться с другими языками программирования, самим попробовать разработать программы, которые можно использовать на уроках и во внеурочной деятельности. А также знание языка Python облегчает написание программы и экономит время при сдаче ЕГЭ по информатике. Данная программа позволяет реализовать эти желания, так как уделяется большое внимание практической работе обучающихся на компьютере, самостоятельной разработке ими программ для решения практических задач.

Основная цель программы – обучение обучающихся программированию посредством языка Python (язык не изучается в качестве базового в школе).

Задачи:

- обучить языку программирования Python и созданию программ на его основе;
- обучить структурному программированию как методу, предусматривающему создание, характерными особенностями которых является: модульность, использование унифицированных структур следования, выбора и повторения, понятных, локально простых и удобочитаемых программ;
- выработать навыки алгоритмизации обучающимися в ее структурном варианте;
- освоение всевозможных методов решения задач;
- развивать алгоритмическое мышление обучающихся;
- формировать навыки грамотной разработки программ.

Описание места курса в учебном плане.

Программа внеурочной деятельности по информатике среднего общего образования рассчитана на 35 часов в год в 10, 11 классах (1 час в неделю).

Перечень учебно-методического обеспечения

1. Учебники «Информатика» для 10-11 классов. Авторы: Л.Л. Босова, А.Ю. Босова. – М.: БИНОМ. Лаборатория знаний.
2. Учебники «Информатика» для 10-11 классов. Авторы: Поляков К.Ю., Еремин Е.А. – М.: БИНОМ. Лаборатория знаний.

Электронные образовательные ресурсы

<http://www.fipi.ru> Портал ФИПИ – Федеральный институт педагогических измерений

<http://www.ege.edu.ru> Портал ЕГЭ (информационной поддержки ЕГЭ)

<http://www.probaege.edu.ru> Портал Единый экзамен

<http://edu.ru/index.php> Федеральный портал «Российское образование»

<https://kpolyakov.spb.ru/school/pycpp.htm> Сайт К. Полякова

<https://lbz.ru/metodist/authors/informatika/3/> Лаборатория знаний Бином (страница Босова Л.Л.)

Способ оценки уровня достижения планируемых результатов: самоконтроль и самооценка, наблюдение, выполнение практических работ, участие в викторинах, конкурсах, олимпиадах.

Личностные, метапредметные и предметные результаты освоения информатики

Основными личностными результатами являются:

- Сформированность навыков сотрудничества со сверстниками, детьми младшего возраста, взрослыми в образовательной, общественно полезной, учебноисследовательской, проектной и других видах деятельности.
- 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; осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов.

Основными метапредметными результатами являются:

- Умение самостоятельно определять цели и составлять планы; самостоятельно осуществлять, контролировать и корректировать учебную и внеучебную (включая внешкольную) деятельность; использовать все возможные ресурсы для достижения целей.
- Готовность и способность к самостоятельной информационно-познавательной деятельности, включая умение ориентироваться в различных источниках информации, критически оценивать и интерпретировать информацию, получаемую из различных источников.
- Владение навыками познавательной рефлексии как осознания совершаемых действий и мыслительных процессов, их результатов и оснований, границ своего знания и незнания, новых познавательных задач и средств их достижения.

Основные предметные результаты отражают:

- Владение навыками алгоритмического мышления и понимание необходимости формального описания алгоритмов.
- Владение умением понимать программы, написанные на выбранном для изучения универсальном алгоритмическом языке; знанием основных конструкций программирования; умением анализировать алгоритмы с использованием таблиц.
- Владение стандартными приёмами написания программы для решения стандартной задачи с использованием основных конструкций программирования.
- Использовать основные понятия, конструкции и структуры данных последовательного программирования, а также правила записи этих конструкций и структур в выбранном для изучения языке программирования;
- Использовать в программах данные различных типов; применять стандартные и собственные подпрограммы для обработки символьных строк; выполнять обработку данных, хранящихся в виде массивов различной размерности; выбирать тип цикла в зависимости от решаемой подзадачи; составлять циклы с использованием заранее определенного инварианта цикла; выполнять базовые операции с текстовыми и двоичными файлами; выделять подзадачи, решение которых необходимо для решения поставленной задачи в полном объеме; реализовывать решения подзадач в виде подпрограмм, связывать подпрограммы в единую программу; использовать модульный принцип построения программ; использовать библиотеки стандартных подпрограмм;
- Применять алгоритмы поиска и сортировки при решении типовых задач;
- Выполнять объектно-ориентированный анализ задачи: выделять объекты, описывать на формальном языке их свойства и методы; реализовывать объектно-ориентированный подход для решения задач средней сложности на выбранном языке программирования;
- Выполнять отладку и тестирование программ в выбранной среде программирования; использовать при разработке программ стандартные библиотеки языка программирования и внешние библиотеки программ; создавать многокомпонентные программные продукты в среде программирования.

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№	Название темы	Количество часов		
		общее	теория	практика
1	Основы языка Python	3	2	1
2	Основные алгоритмические структуры	8	4	4
3	Вспомогательные алгоритмы подпрограммы	11	6,5	4,5
4	Объектно-ориентированное программирование	4	2,5	1,5

5	Приложения с графическим пользовательским интерфейсом	9	4	5
---	---	---	---	---

СОДЕРЖАНИЕ УЧЕБНОГО ПРЕДМЕТА

1. Основы языка Python

Введение в язык программирования Python. История и особенности языка Python. Первый запуск среды разработки.

Структура программы на языке Python. Операции и переменные. Типы данных.

Ввод и вывод данных в программах на языке Python. Считывание данных с клавиатуры. Работа ввода и вывода данных в программах.

2. Основные алгоритмические структуры

Линейные алгоритмы обработки целочисленных данных. Правила записи арифметических выражений. Инструменты обработки целых чисел.

Линейные алгоритмы обработки вещественных чисел. Тип данных float. Инструменты обработки вещественных чисел. Генерация случайных вещественных чисел.

Разветвляющиеся алгоритмы. Логический тип данных. Инструкция ветвления.

Сложные условия. Каскадные ветвления.

Циклические алгоритмы. Цикл с предусловием. Программирование циклов с предусловием.

Цикл с постусловием. Инструкции управления циклом.

Принцип работы цикла с параметром. Программирование цикла с параметром. Функция генерации множества значений из диапазона.

Сложные циклические алгоритмы. Вложенные циклы.

3. Вспомогательные алгоритмы подпрограммы

Этапы решения задачи на компьютере. Последовательное конструирование алгоритма.

Определение функции. Описание функции. Тестирование функции.

Использование функций при решении задач. Повторное использование кода. Модульность.

Определение рекурсии. Рекурсивные алгоритмы. Программирование рекурсивных алгоритмов.

Массивы и списки. Создание, ввод и вывод списков. Перебор элементов списка.

Линейный поиск. Добавление и удаление элементов. Копирование списков.

Алгоритмы сортировки списков. Двоичный поиск элемента в списке.

Определение символьной строки. Методы обработки символьных строк.

Определение матрицы. Реализация матриц на языке Python. Обработка матриц.

Определение исключений. Обработка исключений.

Типы файлов с точки зрения программиста. Программирование файлового ввода/вывода.

4. Объектно-ориентированное программирование

Концепция ООП. Объекты и классы. Понятие ООА.

Программирование объектной модели. Поля и методы класса. Программирование объектов и классов.

Принципы ООП: инкапсуляция. Определение инкапсуляции. Применение инкапсуляции в программировании.

Принципы ООП: наследование и полиморфизм. Определения наследования и полиморфизма.

Применение наследования и полиморфизма в программировании.

5. Приложения с графическим пользовательским интерфейсом

Основы графического интерфейса. Современные прикладные программы. RAD-среды.

Простейшая программа с графическим интерфейсом.

Создание приложения с использованием виджетов. Элементы управления:

кнопки, метки и поля ввода. Создание и настройка окон сообщений.

Создание графического интерфейса без использования программы-визуализатора. Создание окна приложения и запуск цикла обработки событий. Размещение на форме и настройка виджетов.

Управление макетом графического интерфейса: блочный макет. Макет

графического интерфейса. Типы макетов. Блочный макет.

Управление макетом графического интерфейса: сеточный макет. Принцип работы сеточного макета. Программирование сеточного макета. Работа программы при изменении размеров сѐ окна.

Рисование на форме. Рисование при создании графического интерфейса. Инструменты для рисования в PyQt5. Программирование рисования.

Модель обработки данных в приложении с графическим интерфейсом. Декомпозиция приложения с графическим интерфейсом. Модель обработки данных приложения

«Калькулятор».

Представление в приложении с графическим интерфейсом. Представление приложения «Калькулятор». Контроллер в приложении с графическим интерфейсом.

Программирование собственного виджета. Виджет для побитового отображения целых чисел.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ КУРСА

Обучающийся научится:

- использовать понятия языка программирования, среды разработки, набора, запуска и отладки программы;
- использовать среду разработки Wing IDE для создания и запуска программ на языке Python;
- владеть методами пошаговой отладки программ в среде Wing IDE.
- использовать синтаксис языка программирования Python;
- использовать основные типы данных языка программирования Python (int, float, str, bool, list);
- выбирать и использовать структуры данных, необходимых для решения задачи;
- записывать на языке Python арифметические выражения с использованием арифметических операций;
- использовать арифметические операции целочисленного деления с остатком при решении задач;
- использовать основные алгоритмические конструкции языка программирования (арифметическая, условная, циклическая);
- записывать на языке Python логические выражения с использованием логических операций: конъюнкция, дизъюнкция, отрицание, простейшие правила преобразования логических операций и вычислять их значения;
- использовать функции ввода-вывода в языке программирования Python, способы ввода-вывода различных данных;
- владеть методами ввода-вывода данных, чтения последовательности числовых данных неизвестной длины, обработки входной последовательности неизвестной длины без сохранения её в памяти (нахождение среднего арифметического, максимума, второго максимума, количества элемент, равных максимуму и т. д.);
- использовать алгоритмы возведения в степень, быстрого возведения в степень, алгоритм Евклида, алгоритм проверки числа на простоту;
- использовать основные алгоритмы работы с последовательностями данных и массивами (поиск максимума, линейный поиск элемента, сортировки и т. д.);
- владеть методами обработки числовых массивов, уметь реализовывать алгоритмы линейного поиска, сортировки выбором, сортировки вставками, пузырьковой сортировки;
- использовать понятия вспомогательного алгоритма, функции, рекурсии;
- использовать вспомогательные алгоритмы при решении задачи, уметь записывать рекурсивные алгоритмы;
- оперировать с отдельными символами строки и со строками целиком, выполнять операции с символами строк на уровне ASCII-кодов;

- формализовывать условия задачи, создавать модель задачи, разрабатывать алгоритм решения задачи и реализовывать этот алгоритм на языке Python;
- определять результат выполнения заданной программы или её фрагмента;
- выполнять;
- владеть понятием вычислительной сложности алгоритма;
- оценивать вычислительную сложность простых алгоритмов.

Обучающийся получит возможность:

- использовать различные технологии разработки программного обеспечения;
- познакомиться с использованием в программах строковых величин и с операциями со строковыми величинами;
- применять Python не просто для алгоритмических задач, а для решения прикладных задач в тех сферах, где Python оказался наиболее актуален;
- использовать инструменты для профилирования, отладки, тестирования и т.д.;
- проводить анализ данных, как в числовом так и в визуальном формате;
- моделировать проблемы, используя понятия объектно-ориентированного программирования.

КАЛЕНДАРНО-ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

Даты проведения уроков		№ урока	Тема урока
План	Факт		
		1	Введение в язык программирования Python. История и особенности языка Python. Практическая работа № 1. Первый запуск среды разработки.
		2	Структура программы на языке Python. Операции и переменные. Типы данных.
		3	Ввод и вывод данных в программах на языке Python. Считывание данных с клавиатуры. Практическая работа № 2. Работа ввода и вывода данных в программах.
		4	Линейные алгоритмы обработки целочисленных данных. Правила записи арифметических выражений. Инструменты обработки целых чисел. Практическая работа № 3. Линейные алгоритмы обработки целочисленных данных.
		5	Линейные алгоритмы обработки вещественных чисел. Тип данных float. Инструменты обработки вещественных чисел. Генерация случайных вещественных чисел. Практическая работа № 4. Линейные алгоритмы обработки вещественных чисел.
		6	Разветвляющиеся алгоритмы. Логический тип данных. Инструкция ветвления. Практическая работа № 5. Решение задач с использованием условного оператора.
		7	Сложные условия. Каскадные ветвления. Практическая работа № 6. Сложные условия.
		8	Циклические алгоритмы. Цикл с предусловием. Практическая работа № 7. Решение задач с применением данной структуры.
		9	Цикл с постусловием. Инструкции управления циклом. Практическая работа № 8. Решение задач с применением данной структуры.
		10	Принцип работы цикла с параметром. Функция генерации множества значений из диапазона. Практическая работа № 9. Программирование цикла с параметром.
		11	Сложные циклические алгоритмы. Вложенные циклы.

			Практическая работа № 10. Решение задач с применением данной структуры.
		12	Этапы решения задачи на компьютере. Последовательное конструирование алгоритма.
		13	Определение функции. Описание функции. Практическая работа № 11. Тестирование функции.
		14	Использование функций при решении задач. Повторное использование кода. Модульность.
		15	Определение рекурсии. Рекурсивные алгоритмы. Практическая работа № 12. Программирование рекурсивных алгоритмов.
		16	Массивы и списки. Создание, ввод и вывод списков. Перебор элементов списка. Практическая работа № 13. Операции со списками.
		17	Линейный поиск. Практическая работа № 14. Добавление и удаление элементов. Копирование списков.
		18	Алгоритмы сортировки списков. Практическая работа № 15. Двоичный поиск элемента в списке.
		19	Определение символьной строки. Методы обработки символьных строк. Практическая работа № 16. Обработка символьных строк.
		20	Определение матрицы. Реализация матриц на языке Python. Практическая работа № 17. Обработка матриц.
		21	Определение исключений. Практическая работа № 18. Обработка исключений.
		22	Типы файлов с точки зрения программиста. Практическая работа № 19. Программирование файлового ввода/вывода.
		23	Концепция ООП. Объекты и классы. Понятие ООА.
		24	Программирование объектной модели. Поля и методы класса. Программирование объектов и классов. Практическая работа № 20. Программирование объектной модели.
		25	Принципы ООП: инкапсуляция. Определение инкапсуляции. Применение инкапсуляции в программировании. Практическая работа № 21. Инкапсуляция.
		26	Принципы ООП: наследование и полиморфизм. Определения наследования и полиморфизма. Применение наследования и полиморфизма в программировании. Практическая работа № 22. Наследование и полиморфизм
		27	Основы графического интерфейса. Современные прикладные программы. RAD-среды. Практическая работа № 23. Простейшая программа с графическим интерфейсом.
		28	Создание приложения с использованием виджетов. Элементы управления: кнопки, метки и поля ввода. Практическая работа № 24. Создание и настройка окон сообщений.
		29	Создание графического интерфейса без использования программы-визуализатора. Практическая работа № 25. Создание окна приложения и запуск цикла обработки событий. Размещение на форме и настройка виджетов.
		30	Макет графического интерфейса. Типы макетов. Блочный макет. Практическая работа № 26. Управление макетом графического интерфейса: блочный макет.
		31	Принцип работы сеточного макета. Программирование сеточного макета. Работа программы при изменении размеров её окна.

			Практическая работа № 27. Управление макетом графического интерфейса: сеточный макет.
		32	Рисование на форме. Рисование при создании графического интерфейса. Инструменты для рисования в PyQt5. Практическая работа № 28. Программирование рисования.
		33	Модель обработки данных в приложении с графическим интерфейсом. Декомпозиция приложения с графическим интерфейсом. Практическая работа № 29. Модель обработки данных приложения «Калькулятор».
		34	Представление в приложении с графическим интерфейсом. Представление приложения «Калькулятор». Контроллер в приложении с графическим интерфейсом. Практическая работа № 29. Модель обработки данных приложения «Калькулятор».
		35	Итоговая практическая работа № 30. Программирование собственного виджета. Виджет для побитового отображения целых чисел.