

муниципальное бюджетное общеобразовательное учреждение города Ульяновска
«Губернаторский лицей № 101 имени Народного учителя Российской Федерации Ю.И. Латышева»
при ФГБОУ ВО «УлГПУ им. И.Н. Ульянова»

«Рассмотрено»

на заседании

творческих

лабораторий

«24» августа 2021 года протокол № _____

руководитель ТЛ


подпись


расшифровка

«Согласовано»

Заместитель

директора по УВР



подпись



расшифровка

«25» августа 2021 года

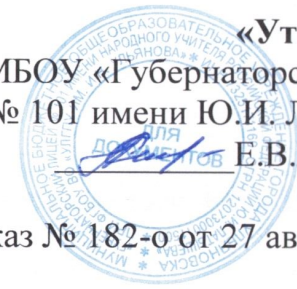
«Утверждаю»

Директор МБОУ «Губернаторский лицей

№ 101 имени Ю.И. Латышева»

 Е.В.Малюгина

Приказ № 182-о от 27 августа 2021



РАБОЧАЯ ПРОГРАММА

учебного предмета «Программирование на языках Python и C++»

уровень образования среднее общее образование. 10-11 класс

Количество часов 260 Срок реализации программы - 2 года

Учитель или группа учителей, разработчиков рабочей программы Нуруллин Наиль Ильдарович, Семак Мария Андреевна, Трунова Виктория Олеговна, Бурлай Ольга Андреевна, Фатахов Азат Асхатович,

Программа разработана в соответствии с федеральным государственным образовательным стандартом среднего общего образования, утвержденного приказом Министерства образования и науки Российской Федерации от 17.05.2012 г. № 413 (с изменениями и дополнениями)

С учетом основной образовательной программы среднего общего образования лицея

С учетом УМК

1. Программирование. Python, C++. Часть 1: учебное пособие/К. Ю. Поляков. -- М.: Бином. Лаборатория знаний, 2019.
2. Программирование. Python, C++. Часть 2: учебное пособие/К. Ю. Поляков. -- М.: Бином. Лаборатория знаний, 2019.
3. Программирование. Python, C++. Часть 3: учебное пособие/К. Ю. Поляков. -- М.: Бином. Лаборатория знаний, 2019.
4. Программирование. Python, C++. Часть 4: учебное пособие/К. Ю. Поляков. -- М.: Бином. Лаборатория знаний, 2019.

1. Планируемые результаты изучения предмета:

10 класс

Личностные результаты

При изучении курса «Программирование на языках Python и C++» в соответствии с требованиями ФГОС формируются следующие личностные результаты:

- сформированность мировоззрения, соответствующего современному уровню развития науки и общественной практики;
- сформированность навыков сотрудничества со сверстниками, детьми младшего возраста, взрослыми в образовательной, общественно полезной, учебно-исследовательской, 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
- бережное, ответственное и компетентное отношение к физическому и психологическому здоровью как к собственному, так и других людей, умение оказывать первую помощь.

Метапредметные результаты

При изучении курса «Программирование на языках Python и C++» в соответствии с требованиями ФГОС формируются следующие метапредметные результаты:

- 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;
- умение продуктивно общаться и взаимодействовать в процессе совместной деятельности, учитывать позиции другого, эффективно разрешать конфликты.
- 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.

Предметные результаты

При изучении курса «Программирование на языках Python и C++» в соответствии с требованиями ФГОС формируются следующие предметные результаты:

Выпускник на углубленном уровне научится:

составлять и отлаживать простые диалоговые программы; узнает особенностей машинных вычислений с целыми и вещественными числами; научится использовать основные алгоритмические конструкции: условные операторы, циклы с условием, циклы по переменной; овладеет методами построения графических изображений программными средствами; овладеет простыми методами программирования компьютерной анимации; познакомится с методами проектирования программ «сверху вниз» и «снизу вверх»; научится использовать вспомогательные алгоритмы (процедуры и функции) для структуризации программ; научится применять рефакторинг для улучшения читаемости программ; научится использовать символные строки; овладеет основными алгоритмами обработки одномерных и двумерных массивов; познакомится с понятием сложности алгоритма; составлять и отлаживать простые диалоговые программы; узнает особенностей машинных вычислений с целыми и вещественными числами; научится использовать основные алгоритмические конструкции: условные операторы, циклы с условием, циклы по переменной; овладеет методами построения графических изображений программными средствами; овладеет простыми методами программирования компьютерной анимации; познакомится с методами

проектирования программ «сверху вниз» и «снизу вверх»; научится использовать вспомогательные алгоритмы (процедуры и функции) для структуризации программ; научится применять рефакторинг для улучшения читаемости программ; научится использовать символьные строки; овладеет основными алгоритмами обработки одномерных и двумерных массивов; познакомится с понятием сложности алгоритма;

Выпускник на углубленном уровне получит возможность научиться:

применять различные алгоритмы сортировки массивов; научится использовать двоичный поиск; научится обрабатывать данные, записанные в текстовые и двоичные файлы, и сохранять в файлах результаты работы программы; научится использовать структуры для объединения данных; научится применять словари, стеки, очереди, деки для решения задач обработки данных; научится использовать деревья для организации данных; познакомится с методами описания графов и некоторыми популярными алгоритмами на графах; научится использовать динамическое программирование для решения комбинаторных и оптимизационных задач; познакомится с понятием выигрышных и проигрышных позиций в играх с полной информацией; познакомится с объектно-ориентированным подходом к разработке программ; научится выполнять объектно-ориентированный анализ задачи, выделять свойства и методы объектов; научится использовать инкапсуляцию для защиты данных объектов; познакомится с понятиями «класс» и «абстрактный класс»; познакомится с понятиями «инкапсуляция», «наследование», «полиморфизм»; научится проектировать несложные иерархии классов для прикладных задач; познакомится с принципами разработки событийно-ориентированных программ; научится создавать программы с графическим интерфейсом на языках Python и C#; научится использовать готовые и создавать новые компоненты (виджеты) для сред быстрой разработки программ; применять различные алгоритмы сортировки массивов; научится использовать двоичный поиск; научится обрабатывать данные, записанные в текстовые и двоичные файлы, и сохранять в файлах результаты работы программы; научится использовать структуры для объединения данных; научится применять словари, стеки, очереди, деки для решения задач обработки данных;

научится использовать деревья для организации данных; познакомится с методами описания графов и некоторыми популярными алгоритмами на графах; научится использовать динамическое программирование для решения комбинаторных и оптимизационных задач; познакомится с понятием выигрышных и проигрышных позиций в играх с полной информацией; познакомится с объектно-ориентированным подходом к разработке программ; научится выполнять объектно-ориентированный анализ задачи, выделять свойства и методы объектов; научится использовать инкапсуляцию для защиты данных объектов; познакомится с понятиями «класс» и «абстрактный класс»; познакомится с понятиями «инкапсуляция», «наследование», «полиморфизм»; научится проектировать несложные иерархии классов для прикладных задач; познакомится с принципами разработки событийно-ориентированных программ; научится создавать программы с графическим интерфейсом на языках Python и C#; научится использовать готовые и создавать новые компоненты (виджеты) для сред быстрой разработки программ.

2. Содержание учебного предмета:

10 класс

Программирование на языке Python (17 часов).

Простейшие программы. Диалоговые программы. Переменные. Консольный ввод и вывод данных. Компьютерная графика. Система координат. Управление пикселями. Графические примитивы: линии, прямоугольники, окружности. Изменение координат. Анимация. Процедуры. Процедуры с параметрами. Рефакторинг. Обработка целых чисел. Арифметические выражения. Деление нацело. Остаток от деления. Обработка вещественных чисел. Особенности представления вещественных чисел в памяти компьютера. Операции с вещественными числами. Случайные и псевдослучайные числа. Генераторы случайных чисел. Ветвления. Условный оператор. Полная и неполная формы условного оператора. Вложенные условные операторы. Логические переменные. Экспертные системы. Сложные условия. Логические операции И, ИЛИ, НЕ. Порядок выполнения операций. Циклы с условием. Алгоритм Евклида. Обработка потока данных. Бесконечные циклы. Циклы по переменной. Шаг изменения переменной цикла. Циклы в компьютерной графике. Узоры. Вложенные циклы. Штриховка.

Программирование на языке C++ (14 часов).

Структура программы на языке C++. Компиляция программы. Препроцессор. Директива *include*. Вывод текста на экран. Диалоговые программы. Ввод и вывод данных. Компьютерная графика. Библиотека *TX Library*. Управление пикселями. Линии и фигуры. Замкнутые фигуры. Процедуры. Процедуры с параметрами. Рефакторинг. Обработка целых чисел. Ограниченность

значений целых чисел. Арифметические выражения. Деление и остаток. Обработка вещественных чисел. Операции с вещественными числами. Случайные и псевдослучайные числа. Ветвления. Условный оператор. Вложенные условные операторы. Логические переменные. Сложные условия. Цикл с предусловием. Циклы с постусловием. Циклы по переменной. Анимация. Обработка нажатия клавиш.

Программирование на языке Python (17 часов).

Этапы создания программ. Методы проектирования программ «сверху вниз» и «снизу вверх». Интерфейс и реализация. Документирование программы. Подпрограммы: процедуры и функции. Процедуры. Процедуры с параметрами. Локальные и глобальные переменные. Функции. Логические функции. Рекурсия. Рекурсивные процедуры и функции. Фракталы. Символьные строки. Сравнение строк. Операции со строками. Обращение к символам. Перебор всех символов. Срезы. Удаление и вставка. Встроенные методы. Поиск в символьных строках. Замена символов. Преобразования «строка — число». Символьные строки в функциях. Рекурсивный перебор. Массивы (списки). Массивы в языке Python. Создание массива. Обращение к элементу массива. Перебор элементов массива. Генераторы. Вывод массива. Ввод массива с клавиатуры. Заполнение массива случайными числами. Алгоритмы обработки массивов. Сумма элементов массива. Подсчёт элементов массива, удовлетворяющих условию. Особенности копирования списков в Python. Поиск в массивах. Линейный поиск. Поиск максимального элемента в массиве. Максимальный элемент, удовлетворяющий условию. Использование массивов в прикладных задачах. Матриц. Создание и заполнение матриц. Вывод матрицы на экран. Перебор элементов матрицы. Квадратные матрицы. Сложность алгоритмов. Асимптотическая сложность.

Программирование на языке C++ (15 часов).

Процедуры. Процедуры с параметрами. Локальные и глобальные переменные. Процедуры, изменяющие аргументы. Рекурсивные процедуры. Построение простых фракталов. Функции в C++. Логические функции. Рекурсивные функции. Символьные строки. Сравнение строк. Сцепление строк. Обращение к символам. Перебор всех символов. Подстрока. Удаление и вставка. Поиск в символьных строках. Замена символов. Преобразования «строка — число». Символьные строки в функциях. Рекурсивный перебор. Массивы в C++. Обращение к элементу массива. Перебор элементов массива. Вывод массива. Ввод массива с клавиатуры. Заполнение массива случайными числами. Алгоритмы обработки массивов. Использование массивов в прикладных задачах. Матрицы. Размещение матрицы в памяти. Заполнение матрицы. Вывод матрицы на экран. Обработка

матриц. Системы управления версиями. Основные приёмы работы с Git. Операции с файлами. Восстановление версии. Работа с удалённым архивом. Ветки. Графические оболочки для Git.

Резерв – 5 часов.

11 класс

Программирование на языке Python (18 часов).

Сортировка массивов. Метод пузырька (сортировка обмeнами). Метод выбора. Сортировка слиянием. Быстрая сортировка. Сортировка в языке Python. Двоичный поиск в массиве данных. Двоичный поиск по ответу. Обработка файлов. Типы файлов. Чтение данных. Запись данных. Обработка данных из файла. Целочисленные алгоритмы. Решето Эратосфена. Целочисленный квадратный корень. Словари. Алфавитно-частотный словарь. Перебор элементов словаря. Структуры. Классы. Создание структур. Работа с полями структур. Хранение структур в файлах. Сортировка структур. Стек. Использование списка. Вычисление арифметических выражений. Скобочные выражения. Системный стек. Очередь. Дек. Деревья. Деревья поиска. Обход дерева. Использование связанных структур. Вычисление арифметических выражений. Графы. Описание графа. Жадные алгоритмы. Минимальное остовное дерево. Алгоритм Дейкстры. Алгоритм Флойда–Уоршелла. Использование списков смежности. Динамическое программирование. Числа Фибоначчи. Количество программ для исполнителя. Двумерные задачи. Поиск оптимального решения. Игровые модели. Выигрышные и проигрышные позиции.

Программирование на языке C++ (14 часов).

Сортировка массивов. Метод пузырька (сортировка простыми обмeнами). Сортировка вставками. Массивы в подпрограммах. Сортировка слиянием. Быстрая сортировка. Стандартная сортировка в языке C++. Двоичный поиск. Обработка файлов. Файловые потоки. Обработка данных из файла. Чтение текстовых файлов по словам. Построчная обработка файлов. Аргументы основной программы. Целочисленные алгоритмы. Решето Эратосфена. «Длинные» числа. Динамические массивы. Тип `vector` из библиотеки STL. Итераторы. Словари. Перебор элементов словаря. Структуры в C++. Обращение к полям структуры. Хранение структур в файлах. Сортировка структур. Стек. Очередь. Хранение очереди в массиве. Дек. Деревья в C++. Обходы дерева. Деревья поиска. Вычисление арифметических выражений. Хранение дерева в массиве. Графы в языке C++. Задача коммивояжёра. Жадные алгоритмы. Случайные перестановки. Передача данных по ссылке. Динамическое программирование. Одномерные задачи. Редактирование строк. Оптимальная стратегия.

Программирование на языке Python (18 часов). Проблема сложности программ. Процедурный и объектно-ориентированный подходы к написанию программ. Классы и объекты. Объектно-ориентированный анализ. Взаимодействие объектов. Свойства и методы. Классы и объекты в программе. Объявление класса. Поля класса. Конструктор класса. Данные и методы класса. Скрытие внутреннего устройства. Доступ к полям через методы. Свойства (*property*). Свойство «только для чтения». Иерархия классов. Наследование. Базовый класс. Доступ к полям. Классы-наследники. Полиморфизм. Разработка модулей. Событийно-ориентированное программирование. Программы с графическим интерфейсом. Форма. Свойства формы. Обработчики событий. Использование компонентов (виджетов). Ввод и вывод данных. Обработка ошибок с помощью исключений. Создание компонентов. Добавление свойств и методов. Составные компоненты. Модель и представление.

Программирование на языке C++ (14 часов). Классы и объекты в языке C++. Объектно-ориентированный анализ задачи. Конструкторы классов. Разбиение на модули. Инкапсуляция. Возможность изменения внутреннего устройства объектов. Свойства «только для чтения». Наследование. Иерархия классов. Базовый класс. Абстрактный класс. «Чистые» виртуальные методы. Защищённые поля и методы (*protected*). Полиморфизм. Указатели на базовый класс. Виртуальные методы. Позднее связывание. Деструктор. Организация взаимодействия объектов. «Умные» указатели. RAD-среды для разработки программ. Язык C# и среда .NET. Проект в C#. Свойства объектов. Обработчики событий. Использование компонентов. Ввод и вывод данных. Обработка ошибок с помощью исключений. Создание новых классов. Статические методы класса. Создание новых компонентов.

3. Тематическое планирование

Воспитательный потенциал данного учебного предмета обеспечивает реализацию следующих целевых приоритетов воспитания обучающихся Среднего общего образования

Основные направления воспитательной деятельности:

1. Гражданское воспитание;
2. Патриотическое воспитание и формирование российской идентичности;
3. Духовное и нравственное воспитание детей на основе российских традиционных ценностей;
4. Приобщение детей к культурному наследию (Эстетическое воспитание);
5. Популяризация научных знаний среди детей (Ценности научного познания);

6. Физическое воспитание и формирование культуры здоровья;
7. Трудовое воспитание и профессиональное самоопределение;
8. Экологическое воспитание

Тематическое планирование 10 класс, всего 68 часов

Номер урока	Тема занятия	Количество часов.	Основные направления воспитательной деятельности*
	Программирование на языке Python		
1.	Первые программы	1	4,7
2.	Диалоговые программы	1	7,8
3.	Компьютерная графика	1	5,1
4.	Процедуры	1	5,6
5.	Обработка целых чисел	1	6
6.	Обработка вещественных чисел	1	6,8
7.	Случайные и псевдослучайные числа	1	4,7
8.	Ветвления	1	7,8
9.	Сложные условия	1	5,1
10.	Циклы с условием	1	5,6
11.	Циклы с условием: практикум	1	6
12.	Анимация	1	6,8

Номер урока	Тема занятия	Количество часов.	Основные направления воспитательной деятельности*
13.	Циклы по переменной	1	4,7
14.	Циклы в компьютерной графике	1	7,8
15.	Выполнение проекта	1	5,1
16.	Выполнение проекта	1	5,6
17.	Выполнение проекта	1	6
	Программирование на языке C++		
18.	Первые программы	1	4,7
19.	Диалоговые программы	1	7,8
20.	Компьютерная графика	1	5,1
21.	Процедуры	1	5,6
22.	Обработка целых чисел	1	6
23.	Обработка вещественных чисел	1	4,7
24.	Ветвления	1	7,8
25.	Циклы	1	5,1
26.	Циклы: практикум	1	5,6
27.	Анимация	1	6
28.	Управление с клавиатуры	1	6,8

Номер урока	Тема занятия	Количество часов.	Основные направления воспитательной деятельности*
29.	Выполнение проекта	1	4,7
30.	Выполнение проекта	1	7,8
31.	Выполнение проекта	1	5,1
Программирование на языке Python			
32.	Проектирование программ	1	4,7
33.	Процедуры	1	7,8
34.	Рекурсия	1	5,1
35.	Функции	1	5,6
36.	Символьные строки	1	6
37.	Обработка символьных строк	1	6,8
38.	Строки в функциях	1	4,7
39.	Массивы	1	7,8
40.	Ввод и вывод массивов	1	5,1
41.	Суммирование элементов массива	1	5,6
42.	Подсчёт элементов массива, удовлетворяющих условию	1	6
43.	Поиск значения в массиве	1	6,8

Номер урока	Тема занятия	Количество часов.	Основные направления воспитательной деятельности*
44.	Поиск максимального элемента в массиве	1	4,7
45.	Игра «Стрельба по тарелкам»	1	7,8
46.	Игра «Стрельба по тарелкам»	1	5,1
47.	Матрицы	1	5,6
48.	Сложность алгоритмов	1	6
	Программирование на языке C++		
49.	Процедуры	1	4,7
50.	Процедуры, изменяющие аргументы	1	7,8
51.	Рекурсия	1	5,1
52.	Функции	1	5,6
53.	Рекурсивные функции	1	6
54.	Символьные строки	1	6,8
55.	Обработка символьных строк	1	4,7
56.	Символьные строки в функциях	1	7,8
57.	Массивы	1	5,1
58.	Игра «Стрельба по тарелкам»	1	5,6
59.	Игра «Стрельба по тарелкам»	1	6

Номер урока	Тема занятия	Количество часов.	Основные направления воспитательной деятельности*
60.	Матрицы	1	6,8
61.	Выполнение проекта	1	4,7
62.	Выполнение проекта	1	7,8
63.	Выполнение проекта	1	5,1
64.	Резерв	1	5,6
65.	Резерв	1	6
66.	Резерв	1	6,8
67.	Резерв	1	4,7
68.	Резерв	1	7,8

*В колонке указывается номер направления в перечне основных направлений воспитательной деятельности через запятую

Тематическое планирование 11 класс, всего 64 часа

Номер урока	Тема занятия	Количество часов.	Основные направления воспитательной деятельности*
	Программирование на языке Python		
1.	Простые алгоритмы сортировки	1	5,1
2.	Сортировка слиянием	1	5,6

Номер урока	Тема занятия	Количество часов.	Основные направления воспитательной деятельности*
3.	Быстрая сортировка	1	6
4.	Двоичный поиск	1	6,8
5.	Обработка файлов	1	4,7
6.	Обработка файлов: практикум	1	7,8
7.	Целочисленные алгоритмы	1	5,1
8.	Словари	1	5,6
9.	Структуры	1	5,1
10.	Структуры: практикум	1	5,6
11.	Стек, очередь, дек	1	6
12.	Деревья	1	6,8
13.	Графы	1	4,7
14.	Графы: практикум	1	7,8
15.	Динамическое программирование	1	5,1
16.	Динамическое программирование: практикум	1	5,6
17.	Игровые модели	1	5,1
18.	Игровые модели: практикум	1	5,6

Номер урока	Тема занятия	Количество часов.	Основные направления воспитательной деятельности*
	Программирование на языке C++		
19.	Простые алгоритмы сортировки	1	5,1
20.	Быстрые алгоритмы сортировки и поиска	1	5,6
21.	Обработка файлов	1	6
22.	Целочисленные алгоритмы	1	6,8
23.	Динамические массивы и словари	1	4,7
24.	Итераторы	1	7,8
25.	Структуры	1	5,1
26.	Структуры: практикум	1	5,6
27.	Стек, очередь, дек	1	5,1
28.	Деревья	1	5,6
29.	Графы	1	6
30.	Графы: практикум	1	6,8
31.	Динамическое программирование	1	4,7
32.	Динамическое программирование: практикум	1	7,8
33.	Резерв	1	5,1

Номер урока	Тема занятия	Количество часов.	Основные направления воспитательной деятельности*
34.	Резерв	1	5,6
	Программирование на языке Python		
35.	Что такое ООП?	1	5,1
36.	Модель задачи: классы и объекты	1	5,6
37.	Классы и объекты в программе	1	6
38.	Классы и объекты в программе: практикум	1	6,8
39.	Скрытие внутреннего устройства	1	4,7
40.	Иерархия классов	1	7,8
41.	Классы-наследники (I)	1	5,1
42.	Классы-наследники (II)	1	5,6
43.	Доработка игры	1	5,1
44.	Событийно-ориентированное программирование	1	5,6
45.	Использование компонентов (виджетов)	1	6
46.	Использование компонентов (виджетов)	1	6,8
47.	Создание компонентов	1	4,7
48.	Модель и представление	1	7,8

Номер урока	Тема занятия	Количество часов.	Основные направления воспитательной деятельности*
49.	Выполнение проекта	1	5,1
50.	Выполнение проекта	1	5,6
Программирование на языке C++			
51.	Классы и объекты	1	5,1
52.	Программа с классами (практикум)	1	5,6
53.	Программа с классами (практикум)	1	6
54.	Инкапсуляция	1	6,8
55.	Наследование	1	4,7
56.	Наследование: практикум	1	7,8
57.	Полиморфизм	1	5,1
58.	Полиморфизм: практикум	1	5,6
59.	Взаимодействие объектов	1	5,1
60.	Простая программа на C#	1	5,6
61.	Использование компонентов	1	6
62.	Ввод и вывод данных	1	6,8
63.	Создание новых классов	1	4,7
64.	Выполнение проекта	1	7,8

Номер урока	Тема занятия	Количество часов.	Основные направления воспитательной деятельности*
65.	Выполнение проекта	1	5,1
66.	Выполнение проекта	1	5,6