

Государственное бюджетное общеобразовательное учреждение гимназия № 49
Приморского района Санкт-Петербурга
197082, Санкт-Петербург, Богатырский пр., д.55, корп.3, лит.А,
Тел./Факс +7 (812) 246-1898, E-mail: info.gim49@obr.gov.spb.ru

«ПРИНЯТА»
Педагогическим советом
ГБОУ гимназии №49
Протокол №1
от «31» августа 2023 года

«УТВЕРЖДЕНА»
Директор гимназии №49
_____/Ф.Ф. Сёмочкина/
Приказ №115 от «31» августа 2023 года

«СОГЛАСОВАНА»
Руководитель отделения
дополнительного образования
_____/В.А Шпинок/
«31» августа 2023 года

**Дополнительная общеобразовательная общеразвивающая программа
Технической направленности
Программирование С++
на 2023-2024 учебный год
год обучения: первый**

Автор-составитель: Чернышева Елена Евгеньевна,
педагог дополнительного образования

Санкт-Петербург
2023

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеобразовательная общеразвивающая программа «**Программирование на языке C++**» разработана в соответствии с нормативно-правовыми документами:

- Федеральным законом «Об образовании в Российской Федерации» от 29.12.2012 года №273-ФЗ (далее – Федеральный закон № 273);

- Приказом Министерства образования и науки Российской Федерации от 29.08. 2013 года № 1008 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;

- Концепцией развития дополнительного образования детей (Распоряжение Правительства РФ от 04.09.2014 года №1726-р);

- Постановлением Главного государственного санитарного врача РФ от 04.07.2014 года №41 «Об утверждении СанПиН 2.4.4.3172-14 «Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей»;

- Письмом Минобрнауки России № 09-3242 от 18.11.2015 года «Методические рекомендации по проектированию дополнительных общеобразовательных общеразвивающих программ»;

- Распоряжением Комитета по образованию Санкт-Петербурга от 01.03.2017 г. № 617-р «Об утверждении Методических рекомендаций по проектированию дополнительных общеразвивающих программ в государственных образовательных организациях Санкт-Петербурга, находящихся в ведении Комитета по образованию»;

- Уставом ГБОУ гимназия № 49 Приморского района Санкт-Петербурга.20. Устава ГБОУ гимназии № 49 Санкт-Петербурга;

Направленность программы: техническая.

Содержание программы предполагает освоение учащимися основ алгоритмизации и программирования, знание национальных приоритетных направлений развития информационных технологий, культурных, исторических особенностей области IT, воспитание гражданственности, чувства патриотизма и ответственности за развитие России.

Программа определяет основные пути развития системного подхода к процессам алгоритмизации и программирования.

Уровень освоения программы: углублённый.

Актуальность.

Достижение цифровой независимости, т. е. возможности сохранения безопасности в сфере IT-решений в рамках всей страны и опоры на собственные разработки, стало одним из важнейших векторов развития России с начала 2022 г. Решение практических задач связано с обучением подрастающего поколения современным, востребованным языкам программирования.

Отличительные особенности программы:

Занятия по алгоритмизации и программированию, направленные решение практических задач способствуют развитию у учащихся профессионального и творческого подхода к работе в IT сфере, готовность к выполнению сложных и ответственных задач в реальных жизненных ситуациях.

Адресат программы:

Программа ориентирована на учащихся 14 -17 лет без специальной подготовки.

Сроки реализации программы:

– Программа рассчитана на 288 часов за 4 года обучения (72 часа в год).

Цель программы – развитие познавательной деятельности учащихся в области новых информационных технологий; совершенствование навыков работы на компьютере и повышение интереса к программированию на языке Python: социальное становление подростков в процессе интеллектуального развития.

Задачи:

Обучающие:

- анализировать поставленную задачу: определять входные данные и результаты, устанавливать связь между ними,
- выделять этапы решения задачи и т.д.;
- разрабатывать алгоритм решения;
- писать текст программы на языке C++;
- отлаживать и тестировать программу;
- готовить описание программы и инструкции для предполагаемых пользователей(документацию);
- дорабатывать и сопровождать программу после отладки и тестирования.

Развивающие:

- формирование активной жизненной позиции,
- формирование морально-волевых качеств, в том числе и в преодолении трудностей,
- содействие удовлетворению интересов подростка в IT области;

Воспитательные:

- воспитание подрастающего поколения на примерах творческого и профессионального подхода к решению актуальных задач в IT области;
- воспитание желания развитию информационных технологий в России,
- развитие инициативы, самостоятельности, навыков работы в команде.

Условия реализации программы

Набор в тренировочные группы воспитанников проходит на добровольной основе, с согласия родителей (представителей), с обязательным допуском врача.

Наполняемость групп:

- 1 год обучения – не менее 15 человек;
- 2 год обучения – не менее 12 человек;
- 3 год обучения – не менее 10 человек;
- 4 год обучения – не менее 10 человек;

Режим занятий:

Занятия групповые. Занятия проводятся 1 раз в неделю по 2 часа.

Формы занятий:

- практические занятия;
- лекции;
- групповая работа над проектами.

Формы организации деятельности учащихся на занятиях:

- индивидуальная;
- групповая;
- индивидуально-групповая.

Материально-техническое обеспечение программы

- Компьютерный класс
- Microsoft Visual Studio
- Eclipse CDT
- Sublime Text
- NetBeans
- QT Creator
- CLion
- CodeLite
- Code::Blocks

Планируемые результаты:

Личностные результаты

У учащихся будут сформированы:

- Интерес к работе в сфере информационных технологий;
- потребность сотрудничества со сверстниками, доброжелательное отношение к сверстникам, бесконфликтное поведение, стремление прислушиваться к мнению одноклассников;

Метапредметными результатами изучения курса является формирование следующих универсальных учебных действий.

Регулятивные УУД:

Учащиеся научатся:

- понимать и принимать учебную задачу, сформулированную учителем;
- осуществлять контроль, коррекцию и оценку результатов своей деятельности.

Познавательная УУД:

Учащиеся научатся:

- проводить сравнение и классификацию объектов;
- понимать и применять полученную информацию при выполнении заданий;
- проявлять индивидуальные творческие способности.

Коммуникативные УУД:

Учащиеся научатся:

- работать в группе, учитывать мнение партнеров, отличные от собственных;
- обращаться за помощью;
- предлагать помощь и сотрудничество;
- слушать собеседника;
- формулировать свои затруднения;
- формулировать собственное мнение и позицию;
- договариваться и приходить к общему решению;
- осуществлять взаимный контроль;

Предметные результаты

1 год обучения

В результате изучения курса на первом уровне учащийся

- 1) научится составлять и отлаживать простые диалоговые программы;
- 2) узнает особенностей машинных вычислений с целыми и вещественными числами;
- 3) научится использовать основные алгоритмические конструкции: условные операторы, циклы с условием, циклы по переменной;
- 4) овладеет методами построения графических изображений программными средствами;
- 5) овладеет простыми методами программирования компьютерной анимации.

2 год обучения

В результате изучения курса на втором уровне учащийся

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

3 год обучения

- 1) научиться применять различные алгоритмы сортировки массивов;
- 2) научиться использовать двоичный поиск;
- 3) научиться обрабатывать данные, записанные в текстовые и двоичные файлы, и сохранять в файлах результаты работы программы;
- 4) научиться использовать структуры для объединения данных;
- 5) научиться применять словари, стеки, очереди, деки для решения задач обработки данных;
- 6) научиться использовать деревья для организации данных;
- 7) познакомиться с методами описания графов и некоторыми популярными алгоритмами на графах;
- 8) научиться использовать динамическое программирование для решения комбинаторных и оптимизационных задач;
- 9) познакомиться с понятием выигрышных и проигрышных позиций в играх с полной информацией;

4 год обучения

- 1) познакомиться с объектно-ориентированным подходом к разработке программ;
- 2) научиться выполнять объектно-ориентированный анализ задачи, выделять свойства и методы объектов;
- 3) научиться использовать инкапсуляцию для защиты данных объектов;
- 4) познакомиться с понятиями «класс» и «абстрактный класс»;
- 5) познакомиться с понятиями «инкапсуляция», «наследование», «полиморфизм»;
- 6) научиться проектировать несложные иерархии классов для прикладных задач;
- 7) познакомиться с принципами разработки событийно-ориентированных программ;
- 8) научиться создавать программы с графическим интерфейсом на языке Python;
- 9) научиться использовать готовые и создавать новые компоненты (виджеты) для сред быстрой разработки программ.

Учебно-тематическое планирование

1 год обучения

№	Тема занятия	Кол-во часов		
		Теория	Практика	Всего
1.	Первые программы	2	2	4
2.	Диалоговые программы	2	4	6
3.	Компьютерная графика	3	9	12
4.	Процедуры	2	2	4
5.	Обработка целых чисел	2	2	4
6.	Обработка вещественных чисел	2	2	4
7.	Ветвления	2	4	6

8.	Циклы	2	4	6
9.	Циклы: практикум	2	4	6
10.	Анимация	2	4	6
11.	Управление с клавиатуры	2	4	6
12.	Выполнение проекта	2	6	8
ВСЕГО		28	44	72

Содержание программы

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.

Учебно-тематическое планирование

2 год обучения

№	Тема урока	Кол-во часов		
		Теория	Практика	Всего
1.	Процедуры	2	2	4
2.	Процедуры, изменяющие аргументы	2	2	4
3.	Рекурсия	2	6	8
4.	Функции	2	6	8
5.	Рекурсивные функции	2	4	6
6.	Символьные строки	1	4	5
7.	Обработка символьных строк	1	2	3
8.	Символьные строки в функциях	2	4	6
9.	Массивы	2	4	6
10.	Игра «Стрельба по тарелкам»	2	4	6
11.	Матрицы	2	6	8
12.	Выполнение проекта	2	6	8
ВСЕГО		28	44	72

Содержание программы

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 сим-волов. 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 обработ-ки 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.

Учебно-тематическое планирование 3 год обучения

№	Тема урока	Кол-во часов		
		Теория	Практика	Всего
1.	Простые алгоритмы сортировки	2	2	4
2.	Быстрые алгоритмы сортировки и поиска	2	4	6
3.	Обработка файлов	2	6	8
4.	Целочисленные алгоритмы	2	6	8
5.	Динамические массивы и словари	2	6	8
6.	Итераторы	2	4	6
7.	Структуры	2	2	4
8.	Структуры: практикум	2	4	6
9.	Стек, очередь, дек	2	4	6
10.	Деревья	2	4	6
11.	Графы	2	4	6
12.	Динамическое программирование	2	2	4
ВСЕГО		24	48	72

Содержание программы

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.

Учебно-тематическое планирование

4 год обучения

№	Тема урока	Кол-во часов		
		Теория	Практика	Всего
1.	Классы и объекты	1	1	2
2.	Программа с классами (практикум)	2	2	4
3.	Программа с классами (практикум)	2	4	6
4.	Инкапсуляция	2	4	6
5.	Наследование	2	2	4
6.	Наследование: практикум	1	2	3
7.	Полиморфизм	2	2	4
8.	Полиморфизм: практикум	2	4	6
9.	Взаимодействие объектов	2	4	6
10.	Простая программа на C#	2	4	6
11.	Использование компонентов	2	6	8
12.	Ввод и вывод данных	2	2	4
13.	Создание новых классов	1	2	3
14.	Выполнение проекта	2	8	10
ВСЕГО		25	47	72

Содержание программы

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.

Календарный учебный график

Год обучения	Дата начала обучения	Дата окончания обучения	Всего учебных недель	Количество учебных часов	Режим занятий
1 год	01.09	25.05	36	72	1 занятие 2 часа в неделю
1 год	01.09	25.05	36	72	1 занятие 2 часа в неделю
1 год	01.09	25.05	36	72	1 занятие 2 часа в неделю
1 год	01.09	25.05	36	72	1 занятие 2 часа в неделю

Оценочные и методические материалы

Оценочные и методические материалы

- словесные (лекции, беседы, работа по устным рекомендациям, анализ проделанной работы);
- наглядные (демонстрация, показ приемов работы, работа по заданиям, просмотр видеоматериалов);
- игровые;
- практические (упражнения);
- решение проблемных задач;
- использования ИКТ (просмотр видеоматериалов, поиск информации в Интернет)

Способы проверки результатов:

- опрос ;
- тестирование;

Тесты и опрос проводятся в течение учебного года. Итоги подводятся в заключительном занятии.

Педагогическая диагностика предполагает:

- личные беседы с детьми и их родителями;
- наблюдение за учащимися во время тренировочных и соревновательных мероприятий;

Формы подведения итогов реализации программы:

- проведение итогового занятия – защита проекта;
- участие в олимпиадах.

Занятия по данной программе состоят из теоретической и практической частей, причем большее количество времени занимает практическая часть, в процессе которой, в основном происходит освоение программного материала. Каждое учебное занятие является звеном системы занятий, связанных в логическую последовательность, построенных друг за другом. Важнейшим требованием современного учебного занятия является обеспечение дифференцированного и индивидуального подхода к обучающимся, с учетом их состояния здоровья, физического развития, особенностей развития психических свойств и качеств.

Содержание программы основывается на следующих основных педагогических принципах образования: демократизации, гуманизации, детоцентризма, природосообразности, культуросообразности, педагогики сотрудничества, дифференциации и индивидуализации.

В зависимости от поставленных педагогических задач, занятия могут быть:

- вводное (введение в предмет, постановка учебных цели и задач, определение средств и методов достижения цели, инструктаж по ТБ)
- занятия – изучение нового материала;
- занятия – повторение, направленные на закрепление, совершенствование ранее освоенных навыков;
- комбинированное (сочетающее изучение, повторение, совершенствование ранее освоенных упражнений) – имеют наибольшее распространение в процессе обучения, - контрольное занятие - проводятся после прохождения части программного материала;
- соревнование, поход, экскурсия и т.д.

Основные принципы построения учебного занятия:

- постепенность в развитии природных данных обучающихся;
- строгая последовательность в изучении и овладении технологических приемов;
- систематичность, регулярность занятий;
- целенаправленность учебного процесса;
- проявление педагогической гибкости по отношению к обучающимся;
- принцип эмоционально-психологической комфортности (создание образовательной среды, обеспечивающей снятие, по возможности, всех стрессообразующих факторов учебного процесса);
- принцип деятельности (новое знание вводится не в готовом виде, а через самостоятельное открытие);
- принцип интеграции разных видов деятельности;
- принцип взаимосвязи и взаимопроникновения программных разделов;
- принцип совместной деятельности педагога, обучающегося и родителей;
- принцип учета индивидуальных особенностей обучающихся.

Основные методы, используемые в учебно-воспитательном процессе.

1. Демонстрационные:

- показ;
- пример;
- видеоиллюстрация.

2. Вербальные:

- объяснение;
- беседа;
- рассказ;
- анализ;
- инструктаж.

3. Практические:

- упражнение;
- игра;
- взаимоконтроль;
- самоконтроль;

4. Стимулирующие:

- соревнование;
- поощрение,

Информационные источники

Для учителя и ученика

1. Поляков К. Учебное пособие «Программирование. Python, C++» 8-11 классы. Профильная школа. 2019г.
2. Васильев А. Н. Самоучитель C++ с примерами и задачами. 4-е издание (переработанное). Книга + виртуальный CD. — СПб.: Наука и Техника, 2016. — 480 с.: ил. (+ виртуальный CD)

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ

**ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ГИМНАЗИЯ № 49 ПРИМОРСКОГО РАЙОНА
САНКТ-ПЕТЕРБУРГА**, Семочкина Фарида Фаридовна, Директор

17.10.23 17:08 (MSK)

Сертификат 3A57D79CE6AE4E1B1876D75D6657705C