Государственное бюджетное учреждение дополнительного образования   
**Центр детского (юношеского) технического творчества Московского района Санкт-Петербурга**

**АННОТАЦИЯ  
к рабочей программе   
дополнительной общеразвивающей программы   
«Программирование на языке C++»**

***Автор - педагог дополнительного образования А.В. Аксенов, И.С. Москвин***

"Язык формирует наш способ мышления и определяет, о чем мы можем мыслить".

Б.Л. Ворф

В последнее время возрос интерес к программированию. Это связано с развитием и внедрением в повседневную жизнь информационно-коммуникационных технологий и, в частности, в учебную деятельность школьников при изучении различных предметов. Для свободной ориентации в информационном потоке подростку надо обладать информационной культурой.

Сейчас, наверное, практически невозможно найти человека, который бы не слышал таких слов, как информатика, программирование, а профессия программиста стала одной из самых многочисленных. Суть деятельности программиста - создание программы для решения задач с помощью компьютера. Для написания программы перед программистом ставится определенная задача, очевидно, что для ее решения недостаточно только знать язык, на котором требуется написать программу, самое главное – найти способ ее решения, алгоритм действий. Программа – это запись алгоритма на каком-либо языке, понятном машине. Процесс перевода на этот язык и называется программированием, а язык, на который осуществляется перевод, называется алгоритмическим языком. Именно поэтому самым важным в профессии программиста является умение создавать хорошие алгоритмы, а, следовательно, и хорошие программы. Для того, чтобы приобщить школьников к высоким информационным технологиям создана дополнительная общеобразовательная общеразвивающая программа **«Программирование на языке C++»** **технической** направленности.

**Актуальность** программы «Программирование на языке C++» заключается в том, что C++ - универсальный язык программирования, задуманный так, чтобы сделать программирование более приятным для серьезного программиста. Этот язык служит двум связанным между собой целям: он дает программисту аппарат для задания действий, которые должны быть выполнены, и формирует концепции, которыми пользуется программист, размышляя о том, что делать. Первой цели идеально отвечает язык, который настолько "близок к машине", что всеми основными машинными аспектами можно легко и просто оперировать достаточно очевидным для программиста образом. С таким умыслом первоначально задумывался **C**. Второй цели идеально отвечает язык, который настолько "близок к решаемой задаче", чтобы концепции ее решения можно было выражать прямо и коротко. С таким умыслом предварительно задумывались средства, добавленные к **C** для создания **C++.**

C++ обычно используется при разработке компиляторов, операционных систем, инструментальных средств. Многие программы, написанные на этом языке, по эффективности близки к программам, написанным на языке ассемблера. В рамках этой программы учащееся освоят принципы структурного программирования, изучат конструкции языка С++, познакомятся с его мощнейшим объектно-ориентированным механизмом.

**Отличительной особенностью** дополнительной программы является то, что в процессе занятий требуется умение пользоваться справочными системами среды программирования. Этот фактор, в частности, приводит к появлению такой межпредметной задачи, как изучение технического английского языка, поскольку все среды программирования являются англоязычными и использование их справочных систем без соответствующего владения английским языком невозможно.

**Адресат программы.** Дополнительная общеобразовательная программа «Программирование на языке C++» рассчитана на учащихся **15-17 лет.** Специальной подготовки для занятий по программе не требуется.

**Цель программы:** реализация интереса подростков к наукоемким технологиям и развитие их технологической культуры через углубленное изучение программирования на языке С++.

**Условия реализации**

Продолжительность обучения **2 года.** Специальной подготовки для поступления в объединения не требуется. Возможен прием обучающихся сразу на 2-й год обучения при условии успешного выполнения вновь поступающим выпускной работы за 1-й год обучения.

Занятия проводятся в группе 1 раз в неделю по 2 часа с чередованием теоретических и практических занятий. Во время теоретической части преподаватель объясняет новый материал, приводит примеры его использования и рассказывает о его особенностях. После этого учащиеся задают появившиеся вопросы. Преподаватель приводит тестовый пример и объясняет его суть учащимся. Они в свою очередь наблюдают результаты выполнения программы на компьютере. Во время практических занятий каждый учащийся выполняет работу за своим компьютером.

**Уровень освоения** – базовый.

**В** конце года проводится самостоятельная работа, возможна защита проектов. В ходе и в конце обучения возможно участие обучающихся в соревнованиях и конкурсах, проводимых внутри учебных групп, между группами, а также в олимпиадах, конференциях, соревнованиях различных уровней, проводимых ЦДЮТТ и другими организациями. Участие обучающегося в таких соревнованиях не только мотивирует его на повышение своего уровня, но и знакомит с другими школами программирования.

**Планируемые результаты обучения**

**Предметные**: знать основные положения математической логики и дискретной математики, понятие алгоритма; основные этапы разработки программы; знать основы алгоритмизации и принципы реализации алгоритмических структур при помощи конструкций языка С++, иметь представление о структуре аппаратной части персонального компьютера, в частности, об устройстве оперативной памяти и внутреннем представлении данных в ней, иметь представление об этапах проектирования программных средств; **уметь** выбирать, согласуясь с особенностями конкретной задачи, и использовать ту или иную парадигму программирования, анализировать приемлемую структуру классов, отлаживать написанный код.

**Личностные**: развивать строгое логическое мышление, основательный подход к решению проблем, формировать широкий взгляд на поиск выхода из нестандартной ситуации. воспитыватьдисциплинированность, усидчивость, стремление к соблюдению хорошего стиля программирования.

**Метапредметные**: познакомиться с основами технического английского языка, развивать способность анализировать результаты своей деятельности, способность поиска выхода из нестандартной ситуации.

**Учебный план 1 года обучения**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Наименование темы** | **Кол-во часов** | | | **Формы контроля** |
| **всего** | **теория** | **практика** |
| **1** | **Введение** | **6** | **4** | **2** |  |
| 1.1 | Общая характеристика языка | 2 | 2 |  | Педагогическое наблюдение |
| 1.2 | Интегрированные среды программирования Turbo C++ 3.0 и C++Builder 6.0 | 2 | 1 | 1 | Контрольные задания. Самоанализ. Наблюдение педагога |
| 1.3 | Этапы решения задач на ПК | 2 | 2 |  | Самоанализ. Наблюдение педагога |
| **2** | **Основы алгоритмизации** | **6** | **4** | **2** |  |
| 2.1 | Понятие алгоритма | 2 | 2 |  | Педагогическое наблюдение, самоанализ |
| 2.2 | Описание алгоритмов | 4 | 2 | 2 | Опрос, самоанализ, педагогическое наблюдение |
| **3** | **Структурное программирование** | **28** | **8** | **20** |  |
| 3.1 | Концепции структурного программирования | 2 | 2 |  | Педагогическое наблюдение |
| 3.2 | Управляющие структуры | 8 | 4 | 4 | Опрос, самоанализ, педагогическое наблюдение |
| 3.3 | Правила написания программ | 2 | 2 |  | Самоанализ, педагогическое наблюдение |
| 3.4 | Применение метода структурного программирования для решения типовых задач | 14 |  | 14 | Опрос, самоанализ, педагогическое наблюдение |
| 3.5 | Контрольное занятие | 2 |  | 2 | Контрольная работа, Взаимооценивание. Наблюдение педагога |
| **4** | **Основы языка Си** | **32** | **16** | **16** |  |
| 4.1 | Синтаксис языка Си | 8 | 4 | 4 | Самоанализ созданных программ, педагогическое наблюдение |
| 4.2 | Концепция данных языка Си | 14 | 8 | 6 | Самоанализ созданных программ, педагогическое наблюдение |
| 4.3 | Организация ввода-вывода | 8 | 4 | 4 | Самоанализ созданных программ, педагогическое наблюдение |
| 4.4 | Контрольное занятие | 2 |  | 2 | Контрольная работа, анализ и самоанализ созданных программ. |
|  | **Всего** | **72** | **32** | **40** |  |

В зависимости от уровня практической подготовки группы педагог имеет право перераспределить часы по темам программы в пределах установленного времени

**Учебный план 2 год обучения**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Наименование темы** | **Кол-во часов** | | | **Формы контроля** |
| **всего** | **теория** | **практика** |
| **1** | **2** | **3** | **4** | **5** |  |
| **1** | **Процедурное программирование** | **18** | **10** | **8** |  |
| 1.1 | Концепции процедурного программирования | 2 | 2 |  | Опрос |
| 1.2 | Функции | 8 | 4 | 4 | Самостоятельная работа. Самоанализ. Педагогическое наблюдение |
| 1.3 | Приемы процедурного программирования | 4 | 2 | 2 | Самостоятельная работа. Самоанализ. Педагогическое наблюдение |
| 1.4 | Правила написания программ | 2 | 2 |  | Педагогическое наблюдение |
| 1.5 | Контрольное занятие | 2 |  | 2 | Взаимооценивание. Контрольная работа. Педагогическое наблюдение |
| **2** | **Модульное программирование** | **4** | **2** | **2** |  |
| 2.1 | Концепции модульного программирования | 4 | 2 | 2 | Анализ выполненных заданий, взаимоанализ |
| **3** | **Объектно-ориентированное программирование** | **36** | **18** | **18** |  |
| 3.1 | Концепции объектно-ориентированного программирования | 2 | 2 |  | Педагогическое наблюдение |
| 3.2 | Класс | 8 | 4 | 4 | Опрос, самостоятельная работа, самоанализ. Педагогическое наблюдение |
| 3.3 | Приемы объектно-ориентированного программирования | 16 | 8 | 8 | Опрос, самостоятельная работа, самоанализ. Педагогическое наблюдение |
| 3.4 | Объектно-ориентированный графический проект | 8 | 2 | 6 | Опрос, самостоятельная работа, самоанализ. Педагогическое наблюдение |
| 3.5 | Контрольное занятие | 2 |  | 2 | Взаимооценивание. Контрольная работа. Педагогическое наблюдение |
| **4** | **Стандарт ANSI** | **14** | **7** | **7** |  |
| 4.1 | Нововведения ANSI-C++ | 12 | 6 | 6 | Контрольная работа, анализ и самоанализ созданных программ |
| 4.2 | Знакомство с библиотекой STL | 2 | 1 | 1 | Анализ и самоанализ созданных программ. Педагогическое наблюдение |
|  | **Всего** | **72** | **37** | **35** |  |

В зависимости от уровня практической подготовки группы педагог имеет право перераспределить часы по темам программы в пределах установленного времени

**Календарный учебный график**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Год обучения | Дата начала обучения | Дата окончания обучения | Всего учебных недель | Количество учебных дней | Количество учебных часов | Режим занятий |
| 1 год | 09.09 | 31.05 | 36 | 26 | 72 | 1 раз в неделю по 2 часа |
| 2 год | 01.09 | 31.05 | 36 | 36 | 72 | 1 раз в неделю по 2 часа |

Государственное бюджетное учреждение дополнительного образования

Центр детского (юношеского) технического творчества Московского района Санкт-Петербурга

|  |  |
| --- | --- |
| **Рассмотрено и принято**  на Педагогическом Совете ЦДЮТТ  Московского района Санкт-Петербурга  Протокол педсовета ЦДЮТТ  № \_\_\_ от\_\_\_\_\_\_\_\_\_\_\_\_\_г. | **Утверждаю**  Директор ЦДЮТТ  Московского района Санкт-Петербурга  \_\_\_\_\_\_\_\_\_\_\_\_\_ Е.А. Исаева  «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_ г. |
|  | Приказ № \_\_\_ от \_\_\_\_\_\_\_\_\_\_\_\_г. |

**Программирование** **на языке С++**

**Рабочая программа**

дополнительной общеразвивающей программы

для подростков 15 – 17 лет

(направленность - техническая)

группа \_\_\_\_\_

Разработчик

Москвин Иван Сергеевич,

педагог дополнительного образования

Санкт – Петербург

Пояснительная записка

C++ - универсальный язык программирования, задуманный так, чтобы сделать программирование более приятным для серьезного программиста. Этот язык служит двум связанным между собой целям: он дает программисту аппарат для задания действий, которые должны быть выполнены, и формирует концепции, которыми пользуется программист, размышляя о том, что делать.

Дополнительная общеобразовательная программа **«Программирование на языке C++»** рассчитана на учащихся 15-17 лет.

**Характеристика группы.** В группе 1 года обучения занимаются подростки, в основном мальчики, 14-16 лет, владеющие навыками работы на компьютере на уровне пользователя, имеющие математические способности, склонность к логическому мышлению и желающие изучить язык программирования, необходимый при будущем обучении в ВУЗе.

**Цель программы:** реализация интереса подростков к наукоемким технологиям и развитие их технологической культуры через углубленное изучение программирования.

**Задачи первого года обучения**

обучающие:

* получить представление об основных конструкциях языка программирования;
* освоить правила составления алгоритмов,
* освоить основные методы и приемы программирования,
* получить представление о профессиях, связанных с программированием;
* научиться формулировать проблему и принимать обоснованные решения этой проблемы;

развивающие задачи:

* планировать и предугадывать возможные нестандартные ситуации, возникающие в процессе создания программ;
* развить навыки логического мышления на основе решения задач на составление алгоритмов;
* развить познавательную активность.

воспитательные задачи:

* воспитать стремление к правильной организации своего рабочего времени через планирование своей работы;
* воспитать, терпение, самостоятельность, целеустремленность.

**Ожидаемые результаты 1 года обучения**

**Предметные.** Учащиеся будутзнать основные положения математической логики и дискретной математики, понятие алгоритма; основные этапы разработки программы; основы алгоритмизации; принципы реализации алгоритмических структур при помощи конструкций языка С++; простые приемы и методы отладки программы; основные понятия объектно-ориентированного программирования (объект, свойства, события). Они будут **иметь представление**: об устройстве оперативной памяти, будут **уметь:** выбирать и использовать ту или иную парадигму программирования, согласуясь с особенностями конкретной задачи; проверять правильность алгоритма, находить и исправлять типовые ошибки; отлаживать написанный программный код

**Личностные.** У них будет развиватьсястрогое логическое мышление, будет воспитыватьсядисциплинированность, усидчивость, терпение, самостоятельность; настойчивость при достижении поставленных задач.

**Метапредметные.** У учащихся будет развиваться способность анализировать результаты своей деятельности, способность поиска выхода из нестандартной ситуации.

**Календарно-тематическое планирование**

**дополнительной общеобразовательной общеразвивающей программы «Программирование на языке С++»**

**группы первого года обучения**

**ПДО Москвин И.С. (группа 101)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **№**  **занят** | **Каб.** | **Содержание занятия** | **Часы** | **Тема** | **Дата факт** | **Подп. зав. отдел** |
|  | 207 | Теория: сравнительная характеристика Си и других языков программирования, пример программы на Си. Инструктаж по охране труда | 2 | 1.1 |  |  |
|  | 207 | Теория: этапы разработки программ с использованием интегрированных сред программирования, главное меню | 2 | 1.2 |  |  |
| Практика: ознакомление с демонстрационной программой, принципами кодирования, отладки и тестирования в средах программирования. Опрос по охране труда |
|  | 207 | Теория: этапы решения задач на ПК: постановка задачи, формализация задачи, разработка алгоритма, программирование, тестирование и отладка, анализ результатов, документирование | 2 | 1.3 |  |  |
|  | 207 | Теория: определение алгоритма, свойства алгоритма, классификация алгоритмов | 2 | 2.1 |  |  |
|  | 207 | Теория: способы описания алгоритмов, основные символы схем алгоритмов | 2 | 2.2 |  |  |
| Практика: примеры описания алгоритма: естественный язык, схема алгоритма |
|  | 207 | Теория: способы описания алгоритмов, элементы структурограмм | 2 |  |  |
| Практика: примеры описания алгоритма: структурограмма, псевдоязык, языки программирования Паскаль и С++ |
|  | 207 | Теория: концепции структурного программирования | 2 | 3.1 |  |  |
|  | 207 | Теория: базовые управляющие структуры: следование, развилка, цикл с предусловием | 2 | 3.2 |  |  |
| Практика: примеры использования базовых и дополнительных структур |
|  | 207 | Теория: базовые управляющие структуры: следование, развилка, цикл с предусловием | 2 |  |  |
| Практика: примеры использования базовых структур |
|  | 207 | Теория: дополнительные управляющие структуры: обход, выбор варианта, цикл с постусловием, цикл с параметром | 2 |  |  |
| Практика: примеры использования дополнительных структур |
|  | 207 | Теория: дополнительные управляющие структуры: обход, выбор варианта, цикл с постусловием, цикл с параметром | 2 |  |  |
| Практика: примеры использования дополнительных структур |
|  | 207 | Теория: рекомендации по стилю программирования, структура программы на Си | 2 | 3.3 |  |  |
|  | 207 | Практика: освоение методов решения типовых задач обработки данных на языке Си | 2 | 3.4 |  |  |
|  | 207 | Практика: освоение методов решения типовых задач обработки данных на языке Си | 2 |  |  |
|  | 207 | Практика: освоение методов решения типовых задач обработки данных на языке Си | 2 |  |  |
|  | 207 | Практика: освоение методов решения типовых задач обработки данных на языке Си | 2 |  |  |
|  | 207 | Практика: освоение методов решения типовых задач обработки данных на языке Си | 2 |  |  |
|  | 207 | Теория: Инструктаж по охране труда  Практика: освоение методов решения типовых задач обработки данных на языке Си. Опрос по охране труда | 2 |  |  |
|  | 207 | Практика: освоение методов решения типовых задач обработки данных на языке Си | 2 |  |  |
|  | 207 | Практика: контроль усвоения материала (контрольная работа) | 2 | 3.5 |  |  |
|  | 207 | Теория: алфавит, лексемы: константы, идентификаторы, служебные и зарезервированные слова, | 2 | 4.1 |  |  |
| Практика: написание тренировочных программ |
|  | 207 | Теория: алфавит, лексемы: константы, идентификаторы, служебные и зарезервированные слова, | 2 |  |  |
| Практика: отладка тренировочных программ |
|  | 207 | Теория: алфавит, лексемы, операции, выражения, операторы, предложения, тексты, препроцессорные директивы | 2 |  |  |
| Практика: написание демонстрационных программ |
|  | 207 | Теория: алфавит, лексемы: операции, выражения, операторы, предложения, тексты, препроцессорные директивы | 2 |  |  |
| Практика: отладка демонстрационных программ |
|  | 207 | Теория: примитивные типы данных языка Си, указатели, массивы | 2 | 4.2 |  |  |
| Практика: написание тренировочных программ |
|  | 207 | Теория: примитивные типы данных языка Си, указатели, массивы, | 2 |  |  |
| Практика: отладка тренировочных программ |
|  | 207 | Теория: динамическая память, строки символов | 2 |  |  |
| Практика: отладка тренировочных программ |
|  | 207 | Теория: агрегаты данных: структуры, объединения, перечисления | 2 |  |  |
| Практика: написание демонстрационных программ |
|  | 207 | Теория: агрегаты данных: структуры, объединения, перечисления | 2 |  |  |
| Практика: написание программ |
|  | 207 | Теория: агрегаты данных: структуры, объединения, перечисления | 2 |  |  |
| Практика: отладка демонстрационных программ |
|  | 207 | Теория: агрегаты данных: структуры, объединения, перечисления | 2 |  |  |
| Практика: отладка демонстрационных программ |
|  | 207 | Теория: консоль, потоки и файлы, стандартные потоки, работа с потоками | 2 | 4.3 |  |  |
| Практика: написание тренировочных программ |
|  | 207 | Теория: консоль, потоки и файлы, стандартные потоки, работа с потоками | 2 |  |  |
| Практика: отладка тренировочных программ |
|  | 207 | Теория: работа с файлами, текстовые файлы, двоичные файлы | 2 |  |  |
| Практика: написание демонстрационных программ |
|  | 207 | Теория: работа с файлами, текстовые файлы, двоичные файлы | 2 |  |  |
| Практика: отладка демонстрационных программ |
|  | 207 | Практика: контроль усвоения материала (контрольная работа) | 2 | 4.4 |  |  |
|  |  |  | **72** |  |  |  |

**Содержание обучения**

**Раздел 1. Введение**

**Тема 1.1 Общая характеристика языка**

Теория: сравнительная характеристика Си и других языков программирования, пример программы на Си. Инструктаж по охране труда.

**Тема 1.2 Интегрированные среды программирования Turbo C++ 3.0 и C++Builder 6.0**

Теория: этапы разработки программ с использованием интегрированных сред программирования, главное меню.

Практика: ознакомление с демонстрационной программой, принципами кодирования, отладки и тестирования в средах программирования. Опрос по охране труда

**Тема 1.3 Этапы решения задач на ПК**

Теория: этапы решения задач на ПК: постановка задачи, формализация задачи, разработка алгоритма, программирование, тестирование и отладка, анализ результатов, документирование

**Раздел 2. Основы алгоритмизации**

**Тема 2.1 Понятие алгоритма**

Теория: определение алгоритма, свойства алгоритма, классификация алгоритмов

**Тема 2.2 Описание алгоритмов**

Теория: способы описания алгоритмов, основные символы схем алгоритмов, элементы структурограмм

Практика: примеры описания алгоритма: естественный язык, схема алгоритма, структурограмма, псевдоязык, языки программирования Паскаль и С++

**Раздел 3. Структурное программирование**

**Тема 3.1 Концепции структурного программирования**

Теория: концепции структурного программирования

**Тема 3.2 Управляющие структуры**

Теория: базовые управляющие структуры: следование, развилка, цикл с предусловием, дополнительные управляющие структуры: обход, выбор варианта, цикл с постусловием, цикл с параметром

Практика: примеры использования базовых и дополнительных структур

**Тема 3.3 Правила написания программ**

Теория: рекомендации по стилю программирования, структура программы на Си

**Тема 3.4 Применение метода структурного программирования для решения типовых задач**

Практика: освоение методов решения типовых задач обработки данных на языке Си

**Тема 3.5 Контрольное занятие**

Практика: контроль усвоения материала (контрольная работа)

**Раздел 4. Основы языка Си**

**Тема 4.1 Синтаксис языка Си**

Теория: алфавит, лексемы: константы, идентификаторы, служебные и зарезервированные слова, операции, выражения, операторы, предложения, тексты, препроцессорные директивы

Практика: написание и отладка демонстрационных и тренировочных программ

**Тема 4.2 Концепция данных языка Си**

Теория: примитивные типы данных языка Си, указатели, массивы, динамическая память, строки символов, агрегаты данных: структуры, объединения, перечисления

Практика: написание и отладка демонстрационных и тренировочных программ

**Тема 4.3 Организация ввода-вывода**

Теория: консоль, потоки и файлы, стандартные потоки, работа с потоками, работа с файлами, текстовые файлы, двоичные файлы

Практика: написание и отладка демонстрационных и тренировочных программ

**Тема 4.4 Контрольное занятие**

Практика: контроль усвоения материала (контрольная работа)

**Ожидаемые результаты 1 года обучения**

**Предметные**

* узнать понятие алгоритма;
* узнать основные этапы разработки программы;
* узнать основы алгоритмизации;
* узнать принципы реализации алгоритмических структур при помощи конструкций языка С++;
* узнать простые приемы и методы отладки программы;
* **узнать** **основные понятия объектно-ориентированного программирования (объект,** свойства, события);
* уметь выбирать и использовать ту или иную парадигму программирования, согласуясь с особенностями конкретной задачи;
* уметь проверять правильность алгоритма, находить и исправлять типовые ошибки;
* уметь отлаживать написанный программный код;
  + иметь представление об устройстве оперативной памяти;

**Личностные**

* у них будет развиваться строгое логическое мышление;
* у них будет развиваться способность анализировать результаты деятельности;
* у них будет воспитываться дисциплинированность, усидчивость, терпение, самостоятельность;
* стремление к самообразованию, к продолжению образования в области программирования

**Межпредметные**

* у учащихся будет развиваться способность анализировать результаты своей деятельности,
* у учащихся будет развиваться способность поиска выхода из нестандартной ситуации.