

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ, НАУКИ И МОЛОДЕЖНОЙ ПОЛИТИКИ
НИЖЕГОРОДСКОЙ ОБЛАСТИ**

**Государственное бюджетное профессиональное образовательное учреждение
«АРЗАМАССКИЙ ТЕХНИКУМ СТРОИТЕЛЬСТВА И ПРЕДПРИНИМАТЕЛЬСТВА»
Центр цифрового образования детей «IT-куб»**

СОГЛАСОВАНО:

Начальник ЦЦОД «IT-Куб»

Г.А. Волгунова

(подпись)

(Ф.И.О)

«16» июня 2021 г.

УТВЕРЖДАЮ:

Директор ГБПОУ АТСП

О.Н. Бабушкин

(подпись)

(Ф.И.О)

Приказ № 29 от «05» 08 2021 г.



**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА**

«Программирование на C-подобных языках»

Направленность – техническая

Возраст обучающихся: 9-12 лет

Срок реализации: 1 год

Объем: 144 часа

Авторы-составители:

Прокопчик Дарья Николаевна,
педагог дополнительного образования,
Лазарева Анастасия Михаловна,
методист

Арзамас

2021

Содержание

1. Комплекс основных характеристик программы	3
1.1 Пояснительная записка.....	3
1.2 Цели и задачи курса.....	10
1.3 Содержание общеразвивающей программы	12
1.3.1 Учебный план.....	12
1.3.2 Содержание учебного плана.....	25
1.4 Перечень планируемых результатов обучения.....	32
2. Комплекс организационно-педагогических условий реализации общеразвивающей программы	33
2.1 Календарный учебный график на 2021-2022 учебный год.....	33
2.2 Условия реализации программы	34
2.3 Формы аттестации и оценочные материалы	36
2.4 Методические материалы	38
Список литературы	42
Приложение	45

1. Комплекс основных характеристик программы

1.1 Пояснительная записка

На сегодняшний день разработка программных продуктов является наиболее востребованным направлением в любых сферах применения.

Для реализации программы выбраны С-подобные языки – одни из популярных языков программирования в современном мире.

Целесообразность изучения данного курса определяется тем, что С-подобные языки работают значительно быстрее, чем другие языки, например Python, который предназначен для решения небольших задач, в которых скорость выполнения не очень важна и применяется для разработки сайтов, решения задач биоинформатики и обработки больших данных. Программы, созданные на С-подобных языках сложнее писать и отлаживать.

В конце обучения будет проведена итоговая аттестация, которая оценит полученные знания обучающихся. Входящая диагностика по программе «Программирование на С-подобных языках» проводится в начале освоения программы с целью выявления первоначального уровня знаний, умений и навыков учащихся.

Программа «Программирование на С-подобных языках» имеет техническую направленность. В системе дополнительного образования программы научно-технической направленности ориентированы на развитие технических и творческих способностей и умений учащихся, на развитие мотивированной и познавательной сфер обучающегося в области информационных технологий, на организацию научно-исследовательской деятельности, что поспособствует в будущем профессиональному самоопределению и освоении информационно-коммуникационных технологий.

Основанием для проектирования и реализации данной общеразвивающей программы служит *перечень следующих нормативных правовых актов и государственных программных документов:*

- Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 N 273-ФЗ;
- Стратегия развития воспитания в Российской Федерации на период до 2025 года. Распоряжение Правительства Российской Федерации от 29 мая 2015 г. № 996-р;
- Письмо Министерства образования и науки РФ от 18.11.2015г. № 09-3242. «О направлении Методических рекомендаций по проектированию дополнительных общеразвивающих программ (включая разноуровневые)»;
- Распоряжение правительства РФ от 04.09. 2014 № 1726-р «Об утверждении Концепции развития дополнительного образования детей»;
- «Основы законодательств РФ об охране здоровья граждан», утвержденные Верховным советом РФ от 22.07.1993 № 5487 - (ред. от 25.11.2009);
- Федеральный закон от 24.07.1998 № 124-ФЗ «Об основных гарантиях прав ребенка в РФ»; Федеральный закон от «Об основах охраны здоровья граждан в Российской Федерации», 2011г.;
- Санитарно-эпидемиологические правила и нормативы СанПиН 2.4.4.3172-14;
- Приказ Министерства просвещения России от 09.11.2018 г. № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» (Приказ №1008 отменен).

Актуальность программы обусловлена развитием современных и перспективных технологий, что позволяет сегодня компьютерам успешно выполнять такие функции, которые в прошлом были им недоступны или выполнялись другими силами и средствами. Благодаря росту возможностей и повышению доступности компьютеров, потенциал использования их в разных сферах экономики стремительно возрастает. Данная образовательная программа позволяет не только научить обучающегося созданию программ, но и подготовить их к планированию и организации работы над разноуров-

невыми техническими проектами. Это в дальнейшем поможет осуществить осознанный выбор вида деятельности в технической области.

Прогностичность дополнительной общеобразовательной общеразвивающей программы «Программирование на С-подобных языках» заключается в том, что она интегрирована и построена с использованием межпредметных связей. Программа объединяет в себе такие направления деятельности как техническое моделирование и проектирование, современные компьютерные технологии. На протяжении образовательной программы обучающиеся работают с оборудованием (Hard skills) и приобретают навыки, которые важны как для участия в командных проектах, так и для жизни в социуме (Soft skills).

Отличительная особенность программы

Дополнительная общеобразовательная общеразвивающая программа «Программирование на С-подобных языках» включает в себя ознакомление с теоретическими аспектами и синтаксисом языка, а также обучаются практическим навыкам программирования на С-подобных языках.

На данную программу обучения принимаются обучающиеся в возрасте 9-12 лет. Такое распределение осуществляется по причине возрастных особенностей обучающихся, а также уже имеющихся знаний и умений, полученных в общеобразовательных учреждениях.

Адресат программы: дополнительная общеобразовательная общеразвивающая программа «Программирование на С-подобных языках» предназначена для обучающихся в возрасте 9-12 лет, не имеющих ограниченных возможностей здоровья, проявляющих интерес к программированию.

Группа формируется **по возрасту** 9-12 лет.

Формы занятий групповые, количество обучающихся в группе – 8-12 человек.

Место проведения занятий: 607220, Нижегородская обл., г. Арзамас, ул. Мира, д. 1.

Возрастные особенности группы

Содержание программы учитывает возрастные психологические особенности обучающихся 9-12 лет, которые определяют выбор форм проведения занятий. На данном этапе ведущей для них становится учебная деятельность. Этот возраст характеризуется тем, что происходит перестройка познавательных процессов: формируется произвольность внимания и памяти, мышление из наглядно-образного преобразуется в словесно-логическое и рассуждающее, формируется способность к созданию умственного плана действий и рефлексии.

Однако игра в этом возрасте продолжает занимать второе по значимости место после учебной деятельности (как ведущей) и существенно влиять на развитие обучающихся. Развивающие игры способствуют самоутверждению, развивают настойчивость, стремление к успеху и другие полезные мотивационные качества, которые могут им понадобиться в их будущей жизни. В таких играх совершенствуется мышление, действия по планированию, прогнозированию, взвешиванию шансов на успех, выбору альтернатив и т. д.

Режим занятий, объем общеразвивающей программы: длительность одного занятия составляет 2 академических часа (по 45 минут) с перерывом (переменой) в 10 минут, периодичность занятий – 2 раза в неделю.

Срок освоения общеразвивающей программы определяется содержанием программы и составляет 1 год (144 часа).

Формы обучения: очная с применением дистанционных образовательных технологий (Закон №273-ФЗ, гл.2, ст.17, п.2.).

Виды занятий: комбинированный, сочетающий в себе элементы теории и практики.

Единицей учебного процесса является блок уроков (раздел). Каждый такой раздел охватывает отдельную информационную технологию или ее часть. Внутри раздела разбивка по времени изучения производится педагогом самостоятельно, но с учетом рекомендованного календарно-тематического плана. С учетом регулярного повторения ранее изученных тем темп изучения отдельных разделов блока определяется субъективными и

объективными факторами.

Каждая тема раздела начинается с постановки задачи - характеристики предметной области или конкретной программы на С-подобных языках, которую предстоит изучить. С этой целью педагог проводит демонстрацию презентации или показывает саму программу, а также готовые работы, выполненные в ней. Закрепление полученных знаний проводится с помощью практики отработки умений самостоятельно решать поставленные задачи, соответствующие минимальному уровню планируемых результатов обучения.

Основные задания являются обязательными для выполнения всеми обучающимися группы. Задания выполняются на компьютере с использованием интегрированной среды разработки. При этом обучающиеся не только формируют новые теоретические и практические знания, но и приобретают новые технологические навыки.

Методика обучения ориентирована на индивидуальный подход. Для того чтобы каждый обучающийся получил наилучший результат обучения, программой предусмотрены индивидуальные задания для самостоятельной работы на компьютере. Такая форма организации обучения стимулирует интерес обучающегося к предмету, активность и самостоятельность обучающихся, способствует объективному контролю глубины и широты знаний, повышению качества усвоения материала обучающимися, позволяет педагогу получить объективную оценку выбранной им тактики и стратегии работы, методики индивидуального обучения и обучения в группе, выбора предметного содержания.

Для самостоятельной работы используются разные по уровню сложности задания, которые носят репродуктивный и творческий характер. Количество таких заданий в работе может варьироваться.

По типу организации взаимодействия педагогов с обучающимися при реализации программы используются личностно ориентированные технологии, технологии сотрудничества.

Реализация программы предполагает использование здоровьесберегающих технологий.

Специальных медицинских противопоказаний к занятиям не существует, но при выборе данного объединения родителям и педагогу необходимо с особым вниманием отнестись к обучающимся, относящимся к группе риска по зрению, так как на занятиях значительное время они пользуются компьютером. Педагогом проводится предварительная беседа с родителями, в которой акцентируется внимание на обозначенную проблему.

Здоровьесберегающая деятельность реализуется:

- созданием безопасных материально-технических условий;
- включением в занятие динамических пауз, периодической смены деятельности обучающихся;
- контролем педагога за соблюдением обучающимися правил работы за персональным компьютером;
- созданием благоприятного психологического климата в учебной группе в целом.

Объем общеразвивающей программы: 144 часа. Форма организации образовательной деятельности – групповая.

Педагогическая целесообразность программы «Программирование на С-подобных языках» заключается в освоении знаний и получении умений, которые позволят обучающимся понять основы языков программирования. Использование различных инструментов развития soft-skills у обучающихся (игропрактика, командная работа) в сочетании с развитием у них hard-компетенций (workshop, tutorial) позволит сформировать у обучающегося целостную систему знаний, умений и навыков.

Аннотация

Программа «Программирование на С-подобных языках» имеет техническую направленность, в ходе которого обучающиеся в доступной форме познакомятся с первоначальными правилами работы программирования, узнают о профессии программиста и специалиста в области информацион-

ных технологий. Обучающиеся получают опыт в области проектирования, создания и выполнения проектов, расширяют свою познавательную сферу.

Обучающиеся изучат основные алгоритмы по созданию компьютерных программ, а также подготовятся к планированию и организации работы над разноуровневыми техническими проектами, что в дальнейшем поможет осуществить осознанный выбор вида деятельности в техническом творчестве. Программа рассчитана на обучающихся 9-12 лет.

1.2 Цели и задачи курса

Целью программы является привлечение обучающихся к исследовательской и изобретательской деятельности, показывая им, что направление интересно и перспективно; развитие творческих способностей и алгоритмического мышления обучающихся изучая основы программирования, что поспособствует в будущем профессиональном самоопределении и освоении информационно-коммуникационных технологий.

Задачи программы:

Обучающие:

- развитие алгоритмических способностей учащихся по программированию;
- формирование умения достаточно самостоятельно решать технические задачи в процессе решения задач;
- формирование знаний, умений и навыков в области программирования на языке C++;
- стимулирование мотивации учащихся к получению знаний, помогать формировать творческую личность обучающегося.

Развивающие:

- развитие у обучающихся интереса к программированию;
- развитие творческого подхода к работе и расширение ассоциативных возможностей мышления;
- развитие коммуникативные компетенции: навыков сотрудничества в коллективе, малой группе (в паре), участия в беседе, обсуждении;
- освоение «hard» и «soft» компетенций;
- поддержание самостоятельности в учебно-познавательной деятельности;
- формирование технического мышления и творческого подхода к работе.

Воспитательные:

- привитие чувства уважения к собственному труду;

- воспитание аккуратности, целеустремленности и точности;
- воспитание трудолюбия, развитие трудовых умений и навыков;
- формирование способности к продуктивному общению и сотрудничеству со сверстниками и взрослыми в процессе творческой деятельности;
- формирование коммуникативной культуры, внимания и уважения к людям.

1.3 Содержание общеразвивающей программы

1.3.1 Учебный план

Таблица 1

№, п/п	Название раздела, темы	Описание тем	Кол-во часов		
			Всего	Теория	Практика
Раздел 1. Знакомство с основными понятиями языка программирования C++					
1-2	Введение	Знакомство. Правила поведения и техника безопасности в кабинете. Правила работы с компьютером. Язык программирования «C++», особенности и области применения. Написание первой программы.	2	1	1
Раздел 2. Вывод данных и стандартные типы данных C++					
3-4	Вывод данных cout	Вывод данных с помощью оператора cout.	2	1	1

5-6	Целочисленные тип данных	Идентификаторы и константы Комментарии. Знакомство с целочисленным типом данных int.	2	1	1
7-8	Символьный тип данных	Знакомство с символьными типами данных char и wchar_t.	2	1	1
9-10	Вещественный тип данных	Знакомство с вещественными типами данных float, double и long double	2	1	1
11-12	Вывод данных printf() и fprintf()	Форматный вывод данных с использованием функций printf() и fprintf()	2	1	1
13-14	Ввод данных	Ввод данным с помощью оператора cin	2	1	1
15-16	Арифметические операции	Знакомство с арифметическими операциями. Унарные и бинарные арифметические операции.	2	1	1

17-18	Логический тип данных	Знакомство с логическим типом данных bool.	2	1	1
Раздел 3. Условные операторы и циклы					
19-20	Операторы выбора if	Знакомство с синтаксисом условной конструкции if – else if– else.	2	1	1
21-22	Решение задач	Отработка навыков решения задач с оператором выбора if.	2	-	2
23-24	Операторы выбора switch	Знакомство с синтаксисом условной конструкции switch.	2	1	1
25-26	Решение задач	Отработка навыков решения задач с оператором выбора switch.	2	-	2
27-28	Оператор цикла for	Знакомство оператором цикла for: понятие циклов, синтаксис цикла.	2	1	1

29-30	Решение задач	Отработка навыков решения задач с циклом for.	2	-	2
31-32	Оператор цикла while	Знакомство оператором цикла while: синтаксис цикла,.	2	1	1
33-34	Решение задач	Отработка навыков решения задач с циклом while.	2	-	2
35-36	Оператор цикла do	Знакомство оператором цикла do: синтаксис цикла.	2	1	1
37-38	Решение задач	Отработка навыков решения задач с циклом do.	2	-	2
39-40	Вложенные циклы	Знакомство и использование вложенных циклов в программном коде	2	1	1

41-44	Решение задач	Отработка навыков решения задач по всему пройденному разделу.	4	-	4
45-46	Исключения	Знакомство и использование исключений с помощью оператора try - catch	2	1	1
47-48	Решение задач	Отработка навыков решения задач с использованием исключений.	2	-	2
Раздел 4. Операторы перехода					
49-50	Оператор перехода goto	Знакомство и использование оператора безусловного перехода goto.	2	1	1
51-52	Решение задач	Отработка навыков решения задач с использованием оператора перехода goto.	2	-	2
53-54	Оператор выхода из цикла	Знакомство и использование оператора выхода из цикла break.	2	1	1

55-56	Оператор перехода continue	Знакомство и использование оператора перехода к следующей итерации цикла continue.	2	1	1
57-58	Решение задач	Отработка навыков решения задач по всему пройденному разделу.	2	-	2
Раздел 5. Массивы					
59-60	Одномерные статические массивы	Определение и работа со статическими одномерными массивами.	2	1	1
61-62	Датчик случайных чисел	Генерация элементов массива через датчик случайных чисел.	2	1	1
63-64	Сортировка одномерного массива	Различные способы сортировки статического одномерного массива.	2	1	1

65-66	Двумерные статические массивы	Определение и работа со статическими двумерными массивами.	2	1	1
67-68	Решение задач	Отработка навыков решения задач по всему пройденному разделу.	2	-	2
Раздел 6. Функции					
69-70	Функции	Объявление и определение функций	2	1	1
71-72	Передача параметров в функцию	Передача параметров в функцию. Передача параметров по значению и по ссылке.	2	1	1
73-74	Массивы в качестве аргументов функций	Передача в функцию одномерных и двумерных статических массивов.	2	1	1

75-76	Возвращаемое значение	Знакомство и использование оператора возврата из функции return.	2	1	1
77-78	Локальные и глобальные переменные	Знакомство с локальными и глобальными переменными в C++. Области видимости переменных.	2	1	1
79-82	Решение задач	Отработка навыков решения задач по всему пройденному разделу.	4	-	4
Раздел 7. Знакомство с основными понятиями языка программирования C#					
83-84	Введение	Язык программирования «C#», особенности и области применения. Отличие от языка программирования C++. Написание первой программы на языке «C#».	2	1	1
Раздел 8. Переменные и выражения					

85-86	Переменные	Объявление переменных и присваивание им значений. Простые типы. Именованные переменные. Литеральные значения.	2	1	1
87-88	Выражения	Математические операторы. Операторы присваивания. Старшинство операторов. Пространство имен.	2	1	1
89-90	Решение задач	Отработка навыков решения задач по всему пройденному разделу.	2	-	2
Раздел 9. Управление порядком выполнения					
91-92	Булева логика	Булевый тип данных. Логические операторы. Старшинство операторов с дополнениями. Оператор goto.	2	1	1
93-94	Тернарный оператор	Знакомство и использование тернарного оператора ветвления.	2	1	1

95-96	Оператор if и switch	Использование операторов ветвления if и switch.	2	1	1
97-98	Организация циклов	Использование операторов цикла do, while и for. Разбор способов прекращения цикла с помощью операторов break и continue.	2	1	1
99-100	Решение задач	Отработка навыков решения задач по всему пройденному разделу.	2	-	2
Раздел 10. Дополнительные сведения о переменных					
101-102	Преобразование типов	Преобразование переменных из одного типа в другой. Неявное и явное преобразование.	2	1	1
103-104	Сложные типы переменных	Перечислимый тип. Структура.	2	1	1

105-106	Массивы	Работа с одномерными и многомерными массивами. Цикл foreach.	2	1	1
107-108	Действия над строками	Работа со строками. Использование основным команд.	2	1	1
109-110	Решение задач	Отработка навыков решения задач по всему пройденному разделу.	2	-	2
Раздел 11. Функции					
111-112	Функции	Описание и использование функций	2	1	1
113-114	Обмен данными с функцией.	Передача параметров в функцию и возвращение значений из функции. Область действия переменных и другие программные конструкции.	2	1	1

115-116	Решение задач	Отработка навыков решения задач с функциями.	2	-	2
Раздел 12. Проектная деятельность					
117-118	Введение в проектную деятельность	Понятие проекта, проектной деятельности, проектной культуры. Проекты в современном мире.	2	2	-
119-120	Классификация проектов	Классификация проектов по различным критериям. Типология проекта.	2	2	-
121-122	Этапы проекта	Этапы работы над проектом. Определение тематик проектов. Формирование проектных групп. Формирование целей проекта. Содержание портфолио проекта.	2	1	1
123-134	Работа над индивидуальным проектом	Выбор темы индивидуального или группового проекта. Планирование индивидуального или группового проекта. Реализация индивидуального или группового проекта.	12	-	12

135-136	Оформление про-екта	Оформление индивидуального проекта. Технические требования к оформлению про-ектных работ. Изучение требований к составлению презентаций для защиты индиви-дуального проекта	2	1	1
137-138	Требования и под-готовка публично-го выступления	Подготовка к выступлению. Общие рекоменда-ции. Техника подготовки. Психологиче-ский аспект готовности к выступлению. Культура выступления: соблюдение правил этикета, ответы на вопросы, заключительное слово.	2	1	1
139-142	Консультации	Подготовка к защите индивидуального проекта	4	-	4
143-144	Итоговая аттестация		2	-	2
Итого:			144	48	96

1.3.2 Содержание учебного плана

Раздел 1. Знакомство с основными понятиями языка программирования C++

Введение

Теория: Знакомство. Правила поведения и техника безопасности в кабинете. Правила работы с компьютером. Язык программирования «C++», особенности и области применения.

Практика: Написание первой программы.

Раздел 2. Вывод данных и стандартные типы данных C++

2.1 Вывод данных cout

Теория: Вывод данных с помощью оператора cout.

Практика: Решение примеров.

2.2 Целочисленный тип данных

Теория: Идентификаторы и константы Комментарии. Знакомство с целочисленным типом данных int.

Практика: Решение примеров.

2.3 Символьный тип данных

Теория: Знакомство с символьными типами данных char и wchar_t.

Практика: Решение примеров.

2.4 Вещественный тип данных

Теория: Знакомство с вещественными типами данных float, double и long double.

Практика: Решение примеров.

2.5 Вывод данных printf() и fprintf()

Теория: Форматный вывод данных с использованием функций printf() и fprintf()

Практика: Решение примеров.

2.6 Ввод данных

Теория: Ввод данным с помощью оператора `cin`

Практика: Решение примеров.

2.7 Арифметические операции

Теория: Знакомство с арифметическими операциями. Унарные и бинарные арифметические операции.

Практика: Решение примеров.

2.8 Логический тип данных

Теория: Знакомство с логическим типом данных `bool`.

Практика: Решение примеров.

Раздел 3. Условные операторы и циклы

3.1 Операторы выбора `if`

Теория: Знакомство с синтаксисом условной конструкции `if – else if–else`.

Практика: Решение примеров.

3.2 Операторы выбора `switch`

Теория: Знакомство с синтаксисом условной конструкции `switch`.

Практика: Решение примеров.

3.3 Оператор цикла `for`

Теория: Знакомство оператором цикла `for`: понятие циклов, синтаксис цикла.

Практика: Решение примеров.

3.4 Оператор цикла `while`

Теория: Знакомство оператором цикла `while`: синтаксис цикла.

Практика: Решение примеров.

3.5 Оператор цикла `do`

Теория: Знакомство оператором цикла `do`: синтаксис цикла.

Практика: Решение примеров.

3.6 Вложенные циклы

Теория: Знакомство и использование вложенных циклов в программном коде

Практика: Решение примеров.

3.6 Исключения

Теория: Знакомство и использование исключений с помощью оператора try – catch.

Практика: Решение примеров.

Раздел 4. Операторы перехода

4.1 Оператор перехода goto

Теория: Знакомство и использование оператора безусловного перехода goto.

Практика: Решение примеров.

4.2 Оператор выхода из цикла

Теория: Знакомство и использование оператора выхода из цикла break.

Практика: Решение примеров.

4.3 Оператор перехода continue

Теория: Знакомство и использование оператора перехода к следующей итерации цикла continue.

Практика: Решение примеров.

Раздел 5. Массивы

5.1 Одномерные статические массивы

Теория: Определение и работа со статическими одномерными массивами.

Практика: Решение примеров.

5.2 Датчик случайных чисел

Теория: Генерация элементов массива через датчик случайных чисел.

Практика: Решение примеров.

5.3 Сортировка одномерного массива

Теория: Различные способы сортировки статического одномерного массива.

Практика: Решение примеров.

5.3 Двумерные статические массивы

Теория: Определение и работа со статическими двумерными массивами.

Практика: Решение примеров.

Раздел 6. Функции

6.1 Функции

Теория: Объявление и определение функций

Практика: Решение примеров.

6.2 Передача параметров в функцию

Теория: Передача параметров в функцию. Передача параметров по значению и по ссылке.

Практика: Решение примеров.

6.3 Массивы в качестве аргументов функций

Теория: Передача в функцию одномерных и двумерных статических массивов.

Практика: Решение примеров.

6.4 Возвращаемое значение

Теория: Знакомство и использование оператора возврата из функции return.

Практика: Решение примеров.

6.4 Локальные и глобальные переменные

Теория: Знакомство с локальными и глобальными переменными в C++. Области видимости переменных.

Практика: Решение примеров.

Раздел 7. Знакомство с основными понятиями языка программирования C#

Введение

Теория: Язык программирования «C#», особенности и области применения. Отличие от языка программирования C++.

Практика: Написание первой программы на языке «C#».2

Раздел 8. Переменные и выражения

8.1 Переменные

Теория: Объявление переменных и присваивание им значений. Простые типы. Именованные переменные. Литеральные значения.

Практика: Решение примеров.

8.2 Выражения

Теория: Математические операторы. Операторы присваивания. Старшинство операторов. Пространство имен.

Практика: Решение примеров.

Раздел 9. Управлением порядком выполнения

9.1 Булева логика

Теория: Объявление переменных и присваивание им значений. Простые типы. Именованные переменные. Литеральные значения.

Практика: Решение примеров.

9.2 Тринарный оператор

Теория: Знакомство и использование тринарного оператора ветвления.

Практика: Решение примеров.

9.3 Оператор if и switch

Теория: Использование операторов ветвления if и switch.

Практика: Решение примеров.

9.4 Организация циклов

Теория: Использование операторов цикла do, while и for. Разбор способов прекращения цикла с помощью операторов break и continue.

Практика: Решение примеров.

Раздел 10. Дополнительные сведения о переменных

10.1 Преобразование типов

Теория: Преобразование переменных из одного типа в другой. Неявное и явное преобразование.

Практика: Решение примеров.

10.2 Сложные типы переменных

Теория: Перечислимый тип. Структура.

Практика: Решение примеров.

10.3 Массивы

Теория: Работа с одномерными и многомерными массивами. Цикл foreach.

Практика: Решение примеров.

10.4 Действия над строками

Теория: Работа со строками. Использование основным команд.

Практика: Решение примеров.

Раздел 11. Функции

11.1 Функции

Теория: Описание и использование функций

Практика: Решение примеров.

11.2 Обмен данными с функцией.

Теория: Передача параметров в функцию и возвращение значений из функции. Область действия переменных и другие программные конструкции.

Практика: Решение примеров.

Раздел 12. Проектная деятельность

12.1 Введение в проектную деятельность

Теория: Понятие проекта, проектной деятельности, проектной культуры. Проекты в современном мире.

12.2 Классификация проектов

Теория: Классификация проектов по различным критериям. Типология проекта.

12.3 Этапы проекта

Теория: Этапы работы над проектом. Формирование целей проекта. Содержание портфолио проекта.

Практика: Определение тематик проектов. Формирование проектных групп.

12.4 Работа над индивидуальным проектом

Практика: Выбор темы индивидуального или группового проекта. Планирование индивидуального или группового проекта. Реализация индивидуального или группового проекта.

12.5 Оформление проекта

Теория: Технические требования к оформлению проектных работ. Изучение требований к составлению презентаций для защиты индивидуального проекта

Практика: Оформление индивидуального проекта.

12.6 Требования и подготовка публичного выступления

Теория: Подготовка к выступлению. Общие рекомендации. Техника подготовки. Психологический аспект готовности к выступлению. Культура выступления: соблюдение правил этикета, ответы на вопросы, заключительное слово.

Практика: Подготовка к выступлению.

12.7 Консультации

Практика: Подготовка к защите индивидуального проекта.

Итоговая аттестация

Итоговая аттестация реализуется в виде защиты индивидуально-го/группового проекта обучающимися на тему, выбранную ранее или предложенную педагогом.

1.4 Перечень планируемых результатов обучения

Предметные результаты:

- умение составлять и применять алгоритмов различных конструкций, исполнение линейных алгоритмов для знакомых формальных исполнителей;
- приобретение опыта проектной деятельности как особой формы учебной работы;
- знание базовых алгоритмических конструкции;
- приобретение обучающимся знания в области создания программ на языке «C++» и «C#».

Личностные результаты:

- формирование устойчивого интереса к техническому творчеству, мотивация к изучению современных направлений в программировании;
- формирование коммуникативных навыков, умение работать в команде;
- формирование логического и творческого мышления;
- развитие внимания, аккуратности, терпения у обучающихся;
- формирование принципов здоровьесбережения;
- развитие уважительного отношения к своему и чужому труду, бережного отношения к используемому оборудованию.

Метапредметные результаты:

- планирование последовательности шагов для достижения целей;
- умение использовать различные источники информации: интернет, книги и журналы, мнение экспертов;
- умение работать в паре и в коллективе;
- умение формулировать, аргументировать и отстаивать свое мнение.

2. Комплекс организационно-педагогических условий реализации общеразвивающей программы

2.1 Календарный учебный график на 2021-2022 учебный год

Год обучения	сентябрь				октябрь				ноябрь				декабрь				январь				февраль				март				апрель				май				Всего учебных недель/ часов	Всего часов по программе																																																																																																																												
даты	01.09 - 06.09				07.09 - 13.09				14.09 - 20.09				21.09 - 27.09				28.09 – 04.10				05.10 – 11.10				12.10 – 18.10				19.10 – 25.10				26.10 – 01.11				02.11 – 08.11				09.11 – 15.11				16.11 – 22.11				23.11 – 29.11				30.11 – 06.12				07.12 – 13.12				14.12 – 20.12				21.12 – 27.12				28.12 – 03.01				04.01 - 10.01				11.01 – 17.01				18.01 – 24.01				25.01 – 31.01				01.02 – 07.02				08.02 – 14.02				15.02 – 21.02				22.02 – 28.02				01.03 – 07.03				08.03 – 14.03				15.03 – 21.03				22.03 – 28.03				29.03 – 04.04				05.04 – 11.04				12.04 – 18.04				19.04- 25.04				26.04 – 02.05				03.05 – 09.05				10.05 – 16.05				17.05 – 23.05				24.05 – 30.05				31.05					
неделя	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40			теор.	практ.																																																																																																																						
1 год обучения	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	2					36/144	44	100																																																																																																																						

Условные обозначения:		
	Занятия по расписанию	Каникулярный период
	Итоговая аттестация обучающихся	

2.2 Условия реализации программы

Основной организационной формой обучения в ходе реализации данной образовательной программы является занятие. Это форма обеспечивает организационную четкость и непрерывность процесса обучения. Знание педагогом индивидуальных особенностей воспитанников позволяет эффективно использовать стимулирующее влияние коллектива на учебную деятельность каждого обучающегося.

Неоспоримым преимуществом занятия, является возможность соединения фронтальных, групповых и индивидуальных форм обучения.

Формы занятий: беседы, обсуждения, теоретические занятия, практические занятия, метод проектов. Также программа курса включает групповые и индивидуальные формы работы обучающихся (в зависимости от темы занятия).

Материально-техническое обеспечение:

Требования к помещению:

- помещение для занятий, отвечающее требованиям СанПин для учреждений дополнительного образования;
- качественное освещение;
- столы, стулья по количеству обучающихся и 1 рабочее место для педагога.

Оборудование:

- ноутбуки на каждого обучающегося и преподавателя;
- принтер;
- Wi-Fi;
- мультимедийный проектор или интерактивная доска для показа презентаций.

Расходные материалы:

- маркеры;
- бумага писчая;
- шариковые ручки.

Информационное обеспечение:

- операционная система Windows;
- поддерживаемые браузеры: Yandex Browser, Chrome, Firefox, Opera, Safari, Edge;
- среда разработки Visual Studio.

Кадровое обеспечение:

Программа реализуется педагогом дополнительного образования – Прокопчик Дарьей Николаевной.

При реализации программы другим педагогом стоит учитывать, что педагогу необходимо познакомиться с технологией обучения программирования на С-подобных языках.

2.3 Формы аттестации и оценочные материалы

Система отслеживания результатов обучающихся выстроена следующим образом:

- определение начального уровня знаний, умений и навыков;
- текущий контроль;
- итоговая аттестация.

Входящая диагностика по программе «Программирование на С-подобных языках» проводится в начале освоения программы с целью выявления первоначального уровня знаний, умений и навыков обучающихся. Обучающемуся задается ряд общих вопросов, касающихся области алгоритмизации, информатики, программирования и информационных технологий в целом. Вопросы подготавливает педагог, примерный список вопросов представлен в Приложении.

Текущий контроль осуществляется путем наблюдения, определения качества выполнения заданий, отслеживания динамики развития обучающегося. Способы проверки уровня освоения тем: опрос, выполнение упражнений, наблюдение. Формы проведения итогов по каждой теме и каждому разделу общеразвивающей программы соответствуют целям и задачам ДООП.

В конце освоения дополнительной общеобразовательной общеразвивающей программы обучающиеся реализуется итоговая аттестация в виде защиты индивидуальных/групповых проектов. Темы для проектов обучающиеся выбирают на свое усмотрение при согласовании с педагогом или получают одну из тем проекта, подготовленных педагогом. Список предлагаемых тем проектов для выполнения обучающимися согласуется с учебной частью не позднее чем за 3 месяца до проведения итоговой аттестации по дополнительной общеобразовательной общеразвивающей программе «Программирование на С-подобных языках».

Индивидуальный/групповой проект оценивается формируемой комиссией по 100-балльной шкале, которая переводится в один из уровней освоения образовательной программы согласно таблице 3:

Таблица 3

Баллы, набранные обучающимся	Уровень освоения
0–30 баллов	низкий
31–70 баллов	средний
71–100 баллов	высокий

Критерии оценивания индивидуального проекта представлены в Приложении.

Состав комиссии (не менее 3 человек): в обязательном порядке входит педагог; приветствуется привлечение ИТ - профессионалов, представителей высших и других учебных заведений, администрации учебной организации.

Компонентами оценки индивидуального проекта являются (по мере убывания значимости): качество ИП, отзыв педагога, уровень презентации и защиты проекта. Если проект выполнен группой учащихся, то при оценивании учитывается не только уровень исполнения проекта в целом, но и личный вклад каждого из авторов. Решение принимается коллегиально.

2.4 Методические материалы

В образовательном процессе используются следующие методы:

- словесный – беседа, рассказ, объяснение, пояснение, вопросы;
- комбинированный – при создании изображения используются несколько графических техник;
- проектно-исследовательский;
- наглядный: использование технических средств, практические задачи, анализ и решение проблемных ситуаций и т. д.

Выбор методов обучения осуществляется исходя из анализа уровня готовности обучающихся к освоению содержания раздела, степени сложности материала, типа учебного занятия. На выбор методов обучения значительно влияет персональный состав группы, индивидуальные особенности, возможности и запросы обучающихся.

Образовательный процесс строится на следующих принципах:

- ***Принцип научности*** – его сущность состоит в том, чтобы обучающийся усваивал реальные знания, правильно отражающие действительность, составляющие основу соответствующих научных понятий.
- ***Принцип наглядности*** – наглядные образы способствуют правильной организации мыслительной деятельности обучающегося. Наглядность обеспечивает понимание, прочное запоминание.
- ***Принцип доступности, учета возрастных и индивидуальных особенностей школьников в процессе обучения по программе.*** Предполагает соотнесение содержания, характера и объема учебного материала с уровнем развития, подготовленности обучающихся. Переходить от легкого к трудному, от известного к неизвестному. Но доступность не отождествляется с легкостью. Обучение, оставаясь доступным, сопряжено с приложением серьезных усилий, что приводит к развитию личности.
- ***Принцип осознания процесса обучения*** – данный принцип предполагает необходимость развития у обучающегося рефлексивной позиции: как я узнал новое, как думал раньше. Если обучающийся видит свои достижения, это

укрепляет в нем веру в собственные возможности, побуждает к новым усилиям. И если обучающийся понимает, в чем и почему он ошибся, что еще не получается, то он делает первый шаг на пути к самовоспитанию.

– **Принцип воспитывающего обучения** – обучающая деятельность педагога, как правило, носит воспитывающий характер. Содержание обучения, формы его организации, методы и средства оказывают влияние на формирование личности в целом.

Используются следующие педагогические технологии:

- технология группового обучения;
- технология коллективно-взаимного обучения;
- технология работы с аудио- и видеоматериалами.

Дидактические материалы:

– дидактические материалы по теме занятия, распечатанные на листе формата А4 для выдачи каждому обучающемуся.

Формы обучения:

– **фронтальная** – предполагает работу педагога сразу со всеми обучающимися в едином темпе и с общими задачами. Для реализации обучения используется компьютер педагога с мультимедиа проектором, посредством которых учебный материал демонстрируется на общий экран;

– **коллективная** – это форма сотрудничества, при котором коллектив обучает каждого своего члена и каждый член коллектива активно участвует в обучении своих товарищей по совместной учебной работе;

– **групповая** – предполагает, что занятия проводятся с подгруппой. Для этого группа разделяется на подгруппы не более 6 человек, работа в которых регулируется педагогом;

– **индивидуальная** – подразумевает взаимодействие педагога с одним обучающимся. Как правило данная форма используется в сочетании с фронтальной. Часть занятия (объяснение новой темы) проводится фронтально, затем

обучающиеся выполняют индивидуальные задачи или общие задачи в индивидуальном темпе.

– **дистанционная** – взаимодействие педагога и обучающихся между собой на расстоянии, отражающее все присущие учебному процессу компоненты. Для реализации дистанционной формы обучения весь дидактический материал размещается в свободном доступе в сети Интернет, происходит свободное общение педагога и обучающихся в социальных сетях, по электронной почте, посредством видеоконференции или в общем чате. Кроме того, дистанционное обучение позволяет проводить консультации обучающихся при самостоятельной работе дома. Налаженная система сетевого взаимодействия обучающегося и педагога, позволяет не ограничивать процесс обучения нахождением в учебной аудитории, обеспечить возможность непрерывного обучения в том числе, для часто болеющих или всех обучающихся в период сезонных карантинов (например, по гриппу) и температурных ограничениях посещения занятий.

Формы организации учебного занятия:

В образовательном процессе помимо традиционного учебного занятия используются многообразные формы, которые несут учебную нагрузку и могут использоваться как активные способы освоения обучающимися образовательной программы, в соответствии с возрастом, составом группы, содержанием учебного раздела: беседа, лекция, мастер-класс, практическое занятие, защита проектов, конкурс, викторина, диспут, круглый стол, «мозговой штурм».

Методы воспитания: мотивация, убеждение, поощрение, упражнение, стимулирование, создание ситуации успеха и др.

Педагогические технологии:

- индивидуализации обучения;
- группового обучения;
- коллективного взаимообучения;
- дифференцированного обучения;
- разноуровневого обучения;

- проблемного обучения;
- развивающего обучения;
- дистанционного обучения;
- игровой деятельности;
- коммуникативная технология обучения;
- коллективной творческой деятельности;
- решения изобретательских задач;
- здоровьесберегающая технология.

Дидактические материалы:

Методические пособия, разработанные педагогом с учетом конкретных задач, варианты демонстрационных программ, материалы по терминологии ПО, инструкции по настройке оборудования, учебная и техническая литература. Используются педагогические технологии индивидуализации обучения и коллективной деятельности.

Список литературы

Нормативные документы

- Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 N 273-ФЗ;
- Стратегия развития воспитания в Российской Федерации на период до 2025 года. Распоряжение Правительства Российской Федерации от 29 мая 2015 г. № 996-р;
- Письмо Министерства образования и науки РФ от 18.11.2015г. № 09-3242. «О направлении Методических рекомендаций по проектированию дополнительных общеразвивающих программ (включая разноуровневые)»;
- Распоряжение правительства РФ от 04.09. 2014 № 1726-р «Об утверждении Концепции развития дополнительного образования детей»;
- «Основы законодательств РФ об охране здоровья граждан», утвержденные Верховным советом РФ от 22.07.1993 № 5487 - (ред. от 25.11.2009);
- Федеральный закон от 24.07.1998 № 124-ФЗ «Об основных гарантиях прав ребенка в РФ»;
- Федеральный закон «Об основах охраны здоровья граждан в Российской Федерации», 2011г.;
- Приказ Министерства просвещения России от 09.11.2018 г. № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» (Приказ №1008 отменен);
- Санитарно-эпидемиологические правила и нормативы СанПиН 2.4.4.3172-14.

Перечень учебной литературы, необходимой для освоения курса

Основные источники

1. Огнева, М. В. Программирование на языке C++: практический курс : учебное пособие для бакалавриата и специалитета / М. В. Огнева, Е. В. Кудрина. - Москва : Издательство Юрайт, 2019. — 335 с.

2. Мюллер, Джон Пол, Семпф, Билл, Сфер, Чак. М98 C# для чайников.: Пер. с англ. - СПб.: ООО "Диалектика", 2019. - 608с.

3. Троелсен, Эндрю, Джепикс, Филипп. Язык программирования C# 7 и платформы .NET и .NET Core, 8-е изд. : Пер. с англ. - СПб. : ООО “Диалектика”, 2018. - 1328 с.

Дополнительные источники

1. Sedgewick R. / Седжвик Р. - Algorithms in C++, Third Edition / Фундаментальные алгоритмы на C++ (3-я редакция, в 5-ти частях, 2 книгах)

2. Джексон Н. Классный учитель: Как работать с трудными учениками, сложными родителями и получать удовольствие от профессии / Н. Джексон. – «Альпина Диджитал», 2015. – 332 с.

3. Шилдт, Г. Искусство программирования на C++. СПб.: БХВ-Петербург, 2005, 496 с.

4. Прата, Стивен. Язык программирования C. Лекции и упражнения, 6-е изд. : Пер. с англ. — М. : ООО “И.Д. Вильямс”, 2015. — 928 с.

*Перечень ресурсов информационно-телекоммуникационной сети
«Интернет»*

1. Веб-учебник для начинающих CppStudio: <http://cppstudio.com>
2. Портал о программировании: <https://code-live.ru/tag/cpp-manual>
3. Уроки программирования на языке C++: <https://ravesli.com/uroki-cpp>
4. Полное руководство по языку программирования C# 9.0 и платформе .NET: <https://metanit.com/sharp/tutorial>

Контрольно-измерительные материалы

к дополнительной общеобразовательной общеразвивающей программе

«Программирование на С-подобных языках»

Дополнительной общеобразовательной программой «Программирование на С-подобных языках» предусмотрено проведение:

– Входящая диагностика – оценка исходного уровня знаний, умений и навыков обучающихся перед началом образовательного процесса. Форма – собеседование.

– Текущий контроль – это оценка качества усвоения обучающимися содержания общеобразовательной программы в период обучения. Форма – педагогическое наблюдение, опрос.

– Итоговая аттестация – это оценка качества усвоения обучающимися уровня достижений, заявленных в дополнительной общеобразовательной программе по завершении всего образовательного курса программы. Форма – защита индивидуального/группового проекта.

1. Материалы входящей диагностики

Входящая диагностика реализуется в форме собеседования. Вопросы для собеседования подготавливаются педагогом и могут изменяться. Примерные вопросы для собеседования представлены ниже:

1. Что такое программирование?
2. Что такое программа?
3. Что такое алгоритм?
4. Когда применяют арифметические операторы?
5. Что такое бета-версия?

Примеры ответов к вопросам для опроса:

1	2	3	4	5
Процесс создания программ.	Это конечный набор шагов, которые при следовании им решают какую-то задачу.	Это графическое представление программы. Блок-схема помогает понять логику работы программы или ее части при проектировании.	Арифметические операторы применяются для выполнения арифметических операций над значениями.	Бета-версия – рабочая версия программы, не готовая к финальному запуску.

Критерии оценивания вопросов собеседования.

Каждый вопрос оценивается от 0 до 20 баллов:

0 баллов – ответ неверный.

10 баллов – ответ частично верный или раскрыт не полностью.

20 баллов – ответ верный.

Баллы за каждый вопрос суммируются и переводятся в уровень освоения согласно таблице 5.

Таблица 5

Баллы, набранные обучающимся	Уровень освоения
0–30 баллов	низкий
31–70 баллов	средний
71–100 баллов	высокий

2. Материалы текущего контроля

Текущий контроль осуществляется путём наблюдения, определения качества выполнения заданий, отслеживания динамики развития обучающегося. Способы проверки уровня освоения тем: опрос, выполнение упражнений, наблюдение, оценка выполненных самостоятельных работ.

Примеры вопросов для опросов:

1. Как подключить стандартную библиотеку `iostream`?
2. Сколько аргументов можно передать в функцию? Какая функция используется для определения типа объекта?
3. Сколько параметров можно передать в деструктор?
4. Что называется конструктором?
5. Объект – это

Ключ к опросу:

1. `#include <iostream>`
2. Неограниченное количество
3. Нельзя передавать параметры в деструктор
4. Метод, имя которого совпадает с именем класса и который вызывается автоматически при создании объекта класса
5. Синтаксическая ошибка.
6. Экземпляр класса

3. Материалы итоговой аттестации

Итоговая аттестация осуществляется в форме защиты индивидуальных/групповых проектов.

Темы для проектов обучающиеся выбирают на свое усмотрение при согласовании с педагогом или получают одну из тем проекта, подготовленных педагогом. Список предлагаемых тем проектов для выполнения обучающимися согласуется с учебной частью не позднее чем за 3 месяца до проведения итоговой аттестации по дополнительной общеобразовательной общеразвивающей программе «Программирование на C-подобных языках».

Индивидуальный/групповой проект оценивается формируемой комиссией по 100-балльной шкале, которая переводится в один из уровней освоения образовательной программы согласно таблице 7.

Таблица 7

Баллы, набранные обучающимся	Уровень освоения
0–30 баллов	низкий
31–70 баллов	средний
71–100 баллов	высокий

Состав комиссии (не менее 3 человек): в обязательном порядке входит педагог; приветствуется привлечение ИТ - профессионалов, представителей высших и других учебных заведений, администрации учебной организации.

Компонентами оценки индивидуального/группового проекта являются (по мере убывания значимости): качество ИП, отзыв педагога, уровень презентации и защиты проекта. Если проект выполнен группой учащихся, то при оценивании учитывается не только уровень исполнения проекта в целом, но и личный вклад каждого из авторов. Решение принимается коллегиально.

Критерии оценки индивидуального/группового проекта (критерий Баллы от 0 до максимального (10)):

1. новизна проекта (Максимально 10 баллов);
2. степень завершенности проекта (Максимально 10 баллов);
3. креативность проекта (Максимально 10 баллов);
4. использование современных библиотек (Максимально 10 баллов);
5. наличие структуры проекта (Максимально 10 баллов);
6. сложность алгоритмов, структур данных (Максимально 10 баллов);
7. объем своего кода (Максимально 10 баллов);
8. презентация (Максимально 10 баллов);
9. наличие дизайна (Максимально 10 баллов);
10. наличие комментариев в коде (Максимально 10 баллов).