

Государственное автономное нетиповое образовательное учреждение
Свердловской области «Дворец молодёжи»
Центр цифрового образования «IT-куб»

Принята на заседании
научно-методического совета
ГАНОУ СО «Дворец молодёжи»
Протокол № 5 от 30.05.2024г.

Утверждена директором
ГАНОУ СО «Дворец молодёжи»
А. Н. Слизько
Приказ №663-д от 30.05.2024г.

Дополнительная общеобразовательная общеразвивающая программа
технической направленности

«Базовые навыки программирования на C-подобных языках»

Стартовый уровень

Возраст обучающихся: 12–17 лет
Объём общеразвивающей программы: 108 часов
Срок реализации: 1 год

СОГЛАСОВАНО:
Начальник центра цифрового
образования «IT-куб»
Е.Н. Лянка

Авторы-составители:
Юшков М.И.,
педагог дополнительного
образования,
Погадаева С.Н.,
методист

г. Екатеринбург, 2024 г.

I. Комплекс основных характеристик программы

1. Пояснительная записка

Современное дополнительное образование предоставляет обучающимся возможность получения знаний и навыков работы с различными технологиями и IT-направлениями, помогая подросткам с самоопределением и выбором жизненного пути, формируя прикладные навыки и воспитывая цифровую культуру, способствуя развитию научно-технического прогресса в целом.

Программирование — фундаментальный навык, в основе которого лежат принципы анализа и синтеза. Владение навыками программирования считается одним из факторов цифровой компетентности и позволяет использовать их для решения конкретных задач. Изучение программирования помогает выработать определенный стиль мышления: умение четко формулировать свои мысли, правильно ставить задачу и находить оптимальные пути ее решения, быстро ориентироваться в потоке информации.

Уже долгое время большой популярностью и востребованностью отличаются языки группы C или C-подобные языки, куда входит язык C, C++, C#. Языки группы C используются для программирования станков и роботизированных систем, программирования микроконтроллеров, разработки драйверов, программ, конструирования других языков, WEB-программирования, разработки игр и других направлений, проще говоря являются языками общего назначения. C-подобные языки относятся к востребованным и перспективным языкам разработки, позволяющим не только заложить фундамент знаний будущих программистов и инженеров, но получить востребованные и актуальные навыки, необходимые в учебной, проектной и повседневной деятельности.

Возможна реализация дополнительной общеобразовательной общеразвивающей программы в сетевой форме. ГАНОУ СО «Дворец молодежи» является базовой организацией, организация-участник определяется на основании заключенного договора о сетевой форме реализации программ.

Направленность программы: программа «Базовые навыки программирования на C-подобных языках» имеет техническую направленность.

Программа разработана с учётом требований, следующих нормативных правовых актов и государственных программных документов:

Федеральный закон от 24.07.1998 № 124-ФЗ «Об основных гарантиях прав ребёнка в РФ»;

Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;

Распоряжение Правительства РФ от 31.03.2022 №678-р «Об утверждении Концепции развития дополнительного образования детей до 2030 года»;

Распоряжение Правительства РФ от 29.05.2015 № 996-р «Об утверждении Стратегии развития воспитания в Российской Федерации на период до 2025 года»;

Приказ Министерства Просвещения РФ от № 629 от 27.07.2022 г. «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;

Приказ Министерства Просвещения Российской Федерации от 03.09.2019 г. №467 «Об утверждении Целевой модели развития региональных систем дополнительного образования детей»;

Приказ Министерства труда и социальной защиты Российской Федерации от 22.09.2021 г. № 652н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых»;

Приказ Министерства науки и высшего образования Российской Федерации совместно с Министерством просвещения Российской Федерации от 05.08.2020 г. № 882/391 «Об организации и осуществлении образовательной деятельности при сетевой форме реализации образовательных программ»;

Постановление Главного государственного санитарного врача РФ от 28.09.2020 г. № 28 «Об утверждении СанПиН 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;

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 Свердловской области № 646-РП от 26.10.2018 «О создании в Свердловской области целевой модели развития региональной системы дополнительного образования детей;

Положение о дополнительных общеобразовательных общеразвивающих программах ГАНОУ СО «Дворец молодежи», утвержденное приказом от 14.05.2020 №269-д;

Положение о сетевой форме реализации дополнительных профессиональных программ в государственном автономном нетиповом образовательном учреждении Свердловской области «Дворец молодежи», утвержденное приказом от 25.12.2019 № 756-д;

Актуальность программы обусловлена ее практической значимостью и состоит в том, что она предоставляет возможность освоения языка программирования для последующего решения поставленных технических задач и реализации творческих проектов, так как в ней делается упор не просто на изучение теоретических основ, а на решении конкретных примеров с использованием программирования. Для изучения выбран язык C++, так как в настоящее время данный язык занимает одно из ведущих мест в области системного и прикладного программирования. Данный язык зарекомендовал себя как надежное средство разработки сложных систем и прикладных программ.

C-подобные языки изучаются в школьном курсе информатики, необходимы для решения олимпиадных и конкурсных заданий, а также сдачи ЕГЭ, имеют прикладной характер и могут использоваться для решения

повседневных задач.

Кроме того, языки группы С являются весьма востребованными языками программирования, используемыми профессиональными инженерами во многих сферах IT-индустрии, поэтому знание любого языка данного семейства даже на базовом уровне повышает шансы будущих выпускников на трудоустройство.

Также данная программа является хорошей базой для перехода на более сложные программы обучения. Так, по итогам успешного освоения программы, обучающийся может быть зачислен на другие общеразвивающие программы центра, которые представляют собой более углублённое и профессионально ориентированное изучение языков программирования и конструирования

Отличительная особенность программы заключается в возможности получения обучающимися универсальных компетенций, необходимых при дальнейшем изучении не только информационных технологий. Также в результате изучения парадигмы объектно-ориентированного подхода к программированию происходит формирование базовых знаний и умений для работы с большинством популярных языков и необходимых при освоении других IT-направлений.

Также в программе отдельный раздел выделен на развитие гибких компетенций обучающихся и обучение методикам командного взаимодействия, работы над проектами, поскольку данные навыки приобретают все большее значение в современном обществе, культуре и профессиональной среде.

Адресат программы: дополнительная общеобразовательная общеразвивающая программа «Базовые навыки программирования на C- подобных языках», предназначена для обучающихся в возрасте 12–17 лет.

Количество обучающихся в группе: 10–14 человек. Формы занятий групповые, состав групп постоянный.

Место проведения занятий: г. Екатеринбург, ул. Красных командиров, 11А.

Возрастные особенности группы: Содержание программы учитывает возрастные и психологические особенности обучающихся в возрасте 12–17 лет, которые определяют выбор форм проведения занятий с обучающимися. Подростки данной возрастной группы характеризуются такими процессами, как изменение структуры личности и бурного физического развития. Происходят качественные изменения и в познавательной деятельности, и в личности, и в межличностных отношениях. У каждого эти изменения происходят в разное время. В этом возрасте начинается переход от мышления, основанного на оперировании конкретными представлениями, к теоретическому мышлению, от непосредственной памяти к логической. В 12-14 лет ведущий тип деятельности – референтно значимый, к нему относятся: проектная деятельность (встреча замысла и результата как авторское действие подростка). В 15-17 лет ведущей деятельностью является – учебно-профессиональная деятельность.

12–14 лет – подростковый период. Характерная особенность – личное самосознание, сознательное проявление индивидуальности. Ведущая потребность – самоутверждение. В подростковый период стабилизируются интересы детей. Основное новообразование – становление взрослости как стремление к жизни в обществе взрослых. К основным ориентирам взросления относятся:

15–17 лет – старший подростковый возраст. Завершение физического и психического созревания. Социальная готовность к общественно полезному производительному труду и гражданской ответственности. В отличие от подросткового возраста, где проявление индивидуальности осуществляется благодаря самоидентификации – «кто я», в юношеском возрасте индивидуальность выражается через самопроявление – «как я влияю». Основная задача педагога дополнительного образования в работе с обучающимися в возрасте 15–17 лет сводится к решению противоречия

между готовностью их к полноценной социальной жизни и недопущением отставания от жизни содержания и организации их образовательной деятельности.

Также следует отметить, что для подростков в данном возрасте характерно стремление углублённо понять себя, разобраться в своих чувствах, настроениях, мнениях, отношениях. Это порождает у подростка стремление к самоутверждению, самовыражению (проявления себя в тех качествах, которые он считает наиболее ценными) и самовоспитанию. Эти процессы позволяют положить начало созданию начального профессионального самоопределения обучающихся.

Режим занятий, периодичность и продолжительность занятий: длительность одного занятия составляет 3 академических часа, периодичность занятий – 1 раз в неделю.

Срок освоения общеразвивающей программы: определяется содержанием программы и составляет 1 год.

Форма обучения: очная, возможна реализация очно с применением электронного обучения и дистанционных образовательных технологий (Закон №273-ФЗ, гл.2, ст.17, п.2.).

Объём общеразвивающей программы: общее количество учебных часов, запланированных на весь период обучения, необходимых для освоения программы: 108 академических часов.

По уровню освоения программа общеразвивающая, **стартового уровня.** Она обеспечивает возможность обучения детей с любым уровнем подготовки.

Осваивая программу стартового уровня, обучающиеся изучают базовые принципы программирования, разработки проектов и построения программных продуктов, структуры и устройства компьютеров.

2. Цель и задачи программы

Цель программы: формирование базовых знаний и навыков программирования посредством изучения С-подобных языков.

Задачи:

Обучающие:

- познакомить с основными предметными понятиями программирования и компьютерных наук;
- обучить основным принципам программирования на С-подобных языках программирования;
- сформировать умение практически применять основные конструкции С-подобных языков программирования путем разработки и написания программ различной направленности и сложности;
- познакомить с базовыми конструкциями и принципами объектно-ориентированного программирования.

Развивающие:

- способствовать формированию и развитию навыка работы с различными источниками информации, умения самостоятельно искать, извлекать и отбирать необходимую информацию;
- способствовать развитию умения излагать мысли в чёткой логической последовательности, отстаивать свою точку зрения;
- развить умения планирования, структурирования и разработки проектов, навыков организации и реализации проектной деятельности;
- способствовать формированию интереса к исследовательской и проектной деятельности.

Воспитательные:

- способствовать воспитанию аккуратности при работе с компьютерным оборудованием;
- способствовать развитию основ коммуникативных отношений внутри проектных групп и в коллективе в целом;

– способствовать формированию понимания необходимости организованного и ответственного отношения к учению, труду, другому человеку, его мнению и деятельности;

– способствовать воспитанию упорства в достижении результата.

2. Содержание общеразвивающей программы

Учебный (тематический) план

Таблица 1

№ п/п	Название раздела, темы	Кол-во часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
Раздел 1. Основы программирования. Введение в проектную деятельность		30	10	20	
1.1	Введение в образовательную программу. Инструктаж по ТБ. Что значит быть честным?	3	1	2	Устный опрос.
1.2	Среда разработки. Ввод и вывод данных, переменные и арифметика	3	1	2	Устный опрос, Практическая работа
1.3	Типы данных	3	1	2	Устный опрос, Практическая работа
1.4	Операторы ветвления, условия	3	1	2	Устный опрос, Практическая работа
1.5	Введение в проектную деятельность: организация команды. Анализ существующих проектов. Разработка идей проектов	3	1	2	Устный опрос, Практическая работа
1.6	Циклы	3	1	2	Устный опрос, Практическая работа
1.7	Массивы	3	1	2	Устный опрос, Практическая работа
1.8	Методы	3	1	2	Устный опрос, Практическая работа
1.9	Функции	3	1	2	Устный опрос, Практическая работа
1.10	Проектная деятельность: цели, задачи и результаты проекта. Паспорт проекта. Планирование и управление проектом.	3	1	2	Устный опрос, Практическая работа
Раздел 2. Объектно-ориентированное Программирование. Проектная деятельность		30	8	22	
2.1	Введение в системы управления проектами. Сервисы для совместной работы над проектами	3	1	2	Устный опрос, Практическая работа

2.2	Объектно-ориентированное программирование. Объекты	3	1	2	Устный опрос, Практическая работа
2.3	Классы	3	1	2	Устный опрос, Практическая работа
2.4	Инкапсуляция	3	1	2	Устный опрос, Практическая работа
2.5	Наследование	3	1	2	Устный опрос, Практическая работа
2.6	Полиморфизм	3	1	2	Устный опрос, Практическая работа
2.7	Абстракция	3	1	2	Устный опрос, Практическая работа
2.8	Рекурсия	3	1	2	Устный опрос, Практическая работа
2.9	Проектная деятельность: средства практической реализации проекта	3	0	3	Устный опрос, Практическая работа
2.10	Промежуточный контроль	3	0	3	Выполнение контрольного тестирования
Раздел 3. Основы алгоритмов. Проектная деятельность		30	8	22	
3.1	Проектная деятельность: написание теоретического обоснования проекта	3	0	3	Устный опрос, Практическая работа
3.2	Алгоритмы и сложность	3	1	2	Устный опрос, Практическая работа
3.3	Полный перебор	3	1	2	Устный опрос, Практическая работа
3.4	Оптимизация перебора	3	1	2	Устный опрос, Практическая работа
3.5	Жадные алгоритмы	3	1	2	Устный опрос, Практическая работа
3.6	Динамическое программирование	3	1	2	Устный опрос, Практическая работа
3.7	Рекурсивные алгоритмы	3	1	2	Устный опрос, Практическая работа

3.8	Алгоритмы «Разделяй и властвуй»	3	1	2	Устный опрос, Практическая работа
3.9	Рандомизированные алгоритмы	3	1	2	Устный опрос, Практическая работа
3.10	Проектная деятельность: написание исследования для проекта	3	0	3	Устный опрос, Практическая работа
Раздел 4. Подготовка итоговых проектов		18	1	17	
4.1	Работа над итоговыми проектами	9	0	9	Тестирование проектов
4.2	Инструменты и методы эффективной презентации	6	1	4	Предзащита, презентация проектов
4.3	Итоговое занятие	3	0	3	Защита итоговых проектов
Итого		108	27	81	

Содержание учебного (тематического) плана

Раздел 1. Основы программирования. Введение в проектную деятельность

Тема 1.1. Введение в образовательную программу. Инструктаж по ТБ. Что значит быть честным?

Теория: Введение в образовательную программу, краткий обзор программы. Инструктаж по технике безопасности. Разговор о том, что значит быть честным.

Практика: Выполнение входной диагностики.

Тема 1.2. Среда разработки. Ввод и вывод данных, переменные и арифметика.

Теория: Среда разработки: Освоение основных концепций среды разработки, включая интерфейс и функциональность для создания и отладки программного кода. Рассмотрение команд и инструментов для ввода данных с клавиатуры и вывода на экран, а также методов взаимодействия программы с пользователем. Роль переменных в программировании и их разновидности для хранения и обработки данных. Изучение основных арифметических операций и их применение для математических вычислений в программировании.

Практика: Выполнение заданий, направленных на практическое применение знаний, включающих ввод данных, работу с переменными и выполнение арифметических операций в выбранной среде разработки (например, с использованием IDE для выбранного языка программирования).

Тема 1.3. Типы данных.

Теория: Изучение различных типов данных в программировании и их роли в хранении информации. Рассмотрение основных типов данных, таких как целые числа, числа с плавающей точкой, символы и строки, логические значения и т. д. Обсуждение особенностей каждого типа данных, их представления в памяти компьютера и возможности использования в программировании. Различия между статической и динамической

типизацией данных.

Практика: Выполнение практических заданий по определению типов данных для различных сценариев программирования. Создание программ, использующих различные типы данных для хранения информации и выполнения операций над ними.

Тема 1.4. Операторы ветвления, условия.

Теория: Введение в операторы ветвления и условия в программировании. Обсуждение концепции условного выполнения кода на основе определенных условий. Изучение основных операторов ветвления, таких как if, else, else if, и их синтаксиса. Обсуждение логических операторов (например, &&, ||, !) для составления условий. Рассмотрение примеров использования операторов ветвления для принятия решений в программах.

Практика: Практическое применение операторов ветвления для решения конкретных задач. Создание программ, которые используют условия для выбора различных путей выполнения кода в зависимости от входных данных. Решение задач с использованием операторов ветвления для выполнения конкретных действий в программе в зависимости от установленных условий.

Тема 1.5. Введение в проектную деятельность: организация команды. Анализ существующих проектов. Разработка идей проектов.

Теория: Введение в понятие проектной деятельности и ее значение в области информационных технологий. Обсуждение роли команды в проекте и принципов эффективного взаимодействия между участниками. Анализ существующих проектов в сфере информационных технологий с целью выявления их особенностей, преимуществ и недостатков. Разработка идей новых проектов на основе полученных знаний и анализа рынка.

Практика: Формирование команд для выполнения проектов, распределение ролей и обсуждение стратегий работы. Изучение и анализ различных проектов в области информационных технологий с целью выявления их особенностей и успешных решений. Проведение мозгового

штурма для генерации новых идей проектов и их дальнейшей проработки в команде. Работа над проектами.

Тема 1.6. Циклы.

Теория: Объяснение концепции циклов в программировании и их роль в автоматизации повторяющихся задач. Рассмотрение различных типов циклов, таких как циклы с предусловием и постусловием, их особенностей и области применения. Изучение синтаксиса циклов в выбранном языке программирования, включая ключевые слова и структуры управления.

Практика: Написание программ с использованием циклов для решения конкретных задач. Выполнение практических упражнений, включающих в себя создание циклов различных типов для обработки данных. Проверка понимания и закрепление материала через выполнение задач с использованием циклов в выбранной среде разработки. Работа над проектами.

Тема 1.7. Массивы.

Теория: Определение массивов как структур данных, позволяющих хранить множество элементов одного типа. Рассмотрение основных понятий, таких как индексация, доступ к элементам массива, инициализация и объявление массивов. Обсуждение различных типов массивов и их применение в решении задач.

Практика: Создание программ, использующих массивы для хранения данных различных типов. Работа с индексами массивов, выполнение операций чтения и записи данных. Практические упражнения по работе с массивами для решения конкретных задач, таких как сортировка, поиск элементов, вычисление статистических показателей и т.д. Работа над проектами.

Тема 1.8. Методы.

Теория: Объяснение понятия методов в программировании как способа организации и структурирования кода. Изучение синтаксиса определения методов, передача параметров, возвращаемые значения. Рассмотрение примеров использования методов для выполнения конкретных задач.

Практика: Написание собственных методов для выполнения различных операций и обработки данных. Использование методов для повышения модульности и читаемости кода. Практические упражнения по созданию и использованию методов для решения разнообразных задач программирования. Работа над проектами.

Тема 1.9. Функции.

Теория: Определение функций как основных строительных блоков программирования, предназначенных для выполнения конкретной задачи. Рассмотрение синтаксиса определения функций, передача параметров, возвращаемые значения. Обсуждение основных принципов проектирования функций.

Практика: Создание и использование функций для выполнения различных операций в программах. Работа с аргументами функций, обработка возвращаемых значений. Практические упражнения по написанию функций для решения конкретных задач программирования. Работа над проектами.

Тема 1.10. Проектная деятельность: цели, задачи и результаты проекта. Паспорт проекта. Планирование и управление проектом.

Теория: Ознакомление с основными понятиями проектной деятельности, включая определение целей, задач и ожидаемых результатов проекта. Обсуждение важности паспорта проекта как инструмента планирования и управления проектом. Изучение методов планирования и управления проектом.

Практика: Выполнение практических заданий по формулированию целей и задач проекта, составлению паспорта проекта и разработке плана работы. Обсуждение и анализ существующих проектов с целью извлечения уроков и лучших практик. Работа в команде над созданием идей проектов и их управлением. Работа над проектами.

Раздел 2. Объектно-ориентированное программирование.

Проектная деятельность.

Тема 2.1. Введение в системы управления проектами. Сервисы

для совместной работы над проектами.

Теория: Определение систем управления проектами (СУП) и их роль в организации и управлении проектами. Обсуждение основных функций и возможностей СУП, таких как планирование, управление ресурсами, контроль и отчетность. Рассмотрение различных видов СУП и их применение в различных сферах бизнеса и проектной деятельности. Изучение сервисов для совместной работы над проектами, таких как системы управления версиями, средства коммуникации и совместного доступа к документам.

Практика: Выполнение практических заданий по созданию и управлению проектами в выбранных сервисах, включая составление списков задач, распределение ролей и ответственностей, установку сроков выполнения задач и ведение обсуждений. Работа над проектами.

Тема 2.2. Объектно-ориентированное программирование. Объекты.

Теория: Введение в парадигму объектно-ориентированного программирования (ООП) и объяснение ее основных принципов. Рассмотрение понятия объекта как основной единицы ООП и его отличительных характеристик. Обсуждение основных преимуществ использования ООП в разработке программного обеспечения.

Практика: Опрос обучающихся для оценки их знаний и опыта в области объектно-ориентированного программирования. Практические задания по созданию объектов и работе с ними в выбранном языке программирования. Разработка простых программ, использующих принципы ООП для создания объектов и взаимодействия между ними. Работа над проектами.

Тема 2.3. Классы.

Теория: Изучение понятия класса как основного строительного блока объектно-ориентированного программирования. Рассмотрение структуры класса, его свойств и методов. Обсуждение принципов инкапсуляции и абстракции в контексте классов.

Практика: Опрос обучающихся для выявления их понимания понятия класса и опыта работы с ним. Практические упражнения по созданию классов

для представления различных объектов и сущностей. Разработка программ, использующих классы для моделирования реальных или вымышленных объектов и их взаимодействия. Работа над проектами.

Тема 2.4. Инкапсуляция.

Теория: Объяснение концепции инкапсуляции в объектно-ориентированном программировании. Рассмотрение ее роли в создании надежных и безопасных программных компонентов. Обсуждение принципов доступа к данным и методам класса через модификаторы доступа (public, private, protected).

Практика: Проведение опроса среди обучающихся для оценки их понимания понятия инкапсуляции и опыта работы с ней. Практические задания по созданию классов с использованием модификаторов доступа для обеспечения инкапсуляции данных. Разработка программ, в которых данные класса защищены от прямого доступа извне. Работа над проектами.

Тема 2.5. Наследование.

Теория: Изучение концепции наследования в объектно-ориентированном программировании. Объяснение роли концепции наследования в создании иерархий классов и повторном использовании кода. Рассмотрение основных принципов и правил использования наследования.

Практика: Опрос обучающихся для оценки их знаний и опыта работы с наследованием. Практические задания по созданию иерархий классов с использованием наследования для расширения функциональности программы. Разработка программ, в которых классы наследуют свойства и методы от других классов. Работа над проектами.

Тема 2.6. Полиморфизм.

Практика: Объяснение концепции полиморфизма в объектно-ориентированном программировании. Рассмотрение его разновидностей, в том числе статического и динамического полиморфизма. Обсуждение преимуществ использования полиморфизма для создания гибких и расширяемых программ.

Практика: Опрос обучающихся для оценки их знаний и понимания концепции полиморфизма. Практические упражнения по созданию программ, использующих различные формы полиморфизма, такие как перегрузка функций, переопределение методов, а также использование интерфейсов и абстрактных классов для достижения полиморфного поведения. Работа над проектами.

Тема 2.7. Абстракция.

Практика: Рассмотрение концепции абстракции в объектно-ориентированном программировании. Объяснение ее роли в создании моделей реальных объектов и явлений. Обсуждение принципов выделения общих характеристик и поведения объектов для создания абстрактных классов и интерфейсов.

Практика: Опрос обучающихся для оценки их понимания и опыта работы с абстракциями. Практические задания по созданию абстрактных классов и интерфейсов, для определения общего поведения и характеристик для группы объектов. Разработка программ, использующих абстракции для упрощения и расширения функциональности. Работа над проектами.

Тема 2.8. Рекурсия.

Теория: Введение в концепцию рекурсии в программировании. Объяснение понятия рекурсивных функций и их особенностей. Рассмотрение примеров использования рекурсии для решения задач.

Практика: Проведение опроса среди обучающихся для выявления их знаний и понимания концепции рекурсии. Практические упражнения по написанию рекурсивных функций для решения задач различной сложности. Разработка программ, использующих рекурсивные алгоритмы для обработки данных. Работа над проектами.

Тема 2.9. Проектная деятельность: средства практической реализации проекта.

Практика: Выбор средств для реализации проекта. Работа над проектами.

Тема 2.10. Промежуточный контроль.

Практика: Выполнение обучающимися контрольного тестирования, включающего вопросы по всем пройденным темам. Оценка результатов и обратная связь для обучающихся по результатам тестирования.

Раздел 3. Основы алгоритмов. Проектная деятельность

Тема 3.1. Проектная деятельность: написание теоретического обоснования проекта.

Практика: Практическое занятие по написанию теоретического обоснования проекта с использованием изученных материалов. Работа в группах для разработки концепции, постановки задач и определения решений. Создание черновиков и обсуждение в группе. Коррекция и улучшение теоретического обоснования.

Тема 3.2. Алгоритмы и сложность.

Теория: Введение в понятие алгоритма и его роль в программировании. Обсуждение основных классов алгоритмов и их применение в решении задач. Рассмотрение концепции сложности алгоритмов и способы ее оценки.

Практика: Проведение опроса среди обучающихся для оценки их понимания основных классов алгоритмов. Практическое занятие по реализации алгоритмов на практике. Разбор примеров алгоритмов различной сложности и их анализ. Работа над проектами.

Тема 3.3. Полный перебор.

Теория: Объяснение понятия полного перебора и его применение в решении задач. Рассмотрение принципов работы полного перебора и его ограничений. Обсуждение примеров задач, для которых полный перебор является оптимальным решением.

Практика: Практическое занятие по реализации алгоритмов полного перебора для решения различных задач. Выполнение практических заданий на использование полного перебора в конкретных сценариях. Обсуждение эффективности и сложности реализации полного перебора. Работа над проектами.

Тема 3.4. Оптимизация перебора.

Теория: Рассмотрение методов оптимизации алгоритмов полного перебора. Обсуждение принципов и приемов оптимизации, таких как отсечение неоптимальных вариантов и применение эвристик. Изучение примеров оптимизации алгоритмов полного перебора.

Практика: Практическое занятие по реализации оптимизированных версий алгоритмов полного перебора для решения задач. Анализ эффективности и сложности оптимизированных алгоритмов. Обсуждение сравнения с базовыми версиями алгоритмов и выявление преимуществ и недостатков оптимизации. Работа над проектами.

Тема 3.5. Жадные алгоритмы.

Теория: Введение в жадные алгоритмы и их применение. Обсуждение принципа жадного выбора на каждом шаге и особенностей их работы. Рассмотрение примеров задач, где жадные алгоритмы применяются для поиска оптимального решения.

Практика: Проведение упражнений по написанию и реализации жадных алгоритмов для решения различных задач. Решение практических кейсов, использующих жадные алгоритмы для оптимизации процессов. Анализ эффективности и применимости жадных алгоритмов в конкретных сценариях. Работа над проектами.

Тема 3.6. Динамическое программирование.

Теория: Введение в концепцию динамического программирования и его применение в решении задач. Обсуждение основных принципов работы динамического программирования, включая разбиение задачи на подзадачи и использование уже найденных решений для оптимизации вычислений. Рассмотрение примеров задач, где динамическое программирование может быть применено.

Практика: Проведение практических упражнений по написанию и реализации алгоритмов динамического программирования для решения различных задач. Решение практических кейсов, где требуется оптимизация

процессов с использованием динамического программирования. Анализ эффективности и применимости динамического программирования в различных сценариях.

Тема 3.7. Рекурсивные алгоритмы.

Теория: Введение в понятие рекурсии в программировании и ее применение при разработке алгоритмов. Объяснение основных принципов работы рекурсивных функций, включая базовый случай и шаг рекурсии. Рассмотрение примеров задач, решаемых с помощью рекурсивных алгоритмов.

Практика: Проведение практических занятий по написанию рекурсивных функций для решения различных задач. Разработка программного кода, использующего рекурсивные алгоритмы для обработки данных. Анализ и сравнение эффективности рекурсивных и итеративных решений для одной и той же задачи.

Тема 3.8. Алгоритмы «Разделяй и властвуй».

Теория: Введение в концепцию алгоритмов «Разделяй и властвуй». Обсуждение основных принципов работы таких алгоритмов и их применение в решении различных задач. Рассмотрение примеров задач, для которых алгоритмы «Разделяй и властвуй» являются эффективным решением.

Практика: Опрос обучающихся для проверки их понимания концепции алгоритмов «Разделяй и властвуй» и способов их применения. Практическое занятие, в ходе которого обучающиеся решают задачи с использованием алгоритмов «Разделяй и властвуй». Выполнение практических заданий для отработки применения алгоритмов «Разделяй и властвуй» на конкретных примерах. Работа над проектами.

Тема 3.9. Рандомизированные алгоритмы.

Теория: Введение в рандомизированные алгоритмы и их роль в решении сложных задач. Обсуждение основных принципов работы рандомизированных алгоритмов и их преимуществ. Рассмотрение примеров задач, для которых рандомизированные алгоритмы могут быть полезны.

Практика: Проверка уровня понимания обучающимися концепции рандомизированных алгоритмов через проведение опроса. Практическое занятие по реализации рандомизированных алгоритмов для решения различных задач. Выполнение практических заданий на использование рандомизированных алгоритмов в конкретных сценариях. Работа над проектами.

Тема 3.10. Проектная деятельность: написание исследования для проекта.

Практика: Анализ предыдущих работ и исследований по теме проекта. Подготовка плана исследования, определение целей и задач проекта. Написание теоретического обоснования проекта, включая обзор литературы и описание методологии исследования.

Раздел 4. Подготовка итоговых проектов.

Тема 4.1. Работа над итоговыми проектами.

Практика: Разработка итоговых проектов, тестирование, устранение багов, отладка.

Тема 4.2. Инструменты и методы эффективной презентации.

Теория: Обзор инструментов для создания эффективной презентации и методов подачи информации, взаимодействия с публикой.

Практика: Разработка презентации, доклада, защита (пробное выступление)

Тема 4.3. Итоговое занятие.

Практика: Представление проектов, выступление перед комиссией, защита проектов.

4. Планируемые результаты программы

Предметные результаты:

- знание основных предметных понятий программирования и компьютерных наук;
- способность применения С-подобных языков на практике для решения практических задачи и разработки продуктов;
- знание базовых конструкций и принципов объектно-ориентированного программирования;
- понимание основных принципов программирования на С-подобных языках программирования.

Личностные результаты:

- проявление аккуратности при работе с компьютерным оборудованием;
- сформированы коммуникативные компетентности в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности;
- проявление ответственного отношения к обучению, упорства в достижении результата.

Метапредметные результаты:

- проявление умения искать, извлекать и отбирать нужную информацию из открытых источников;
- проявление умения формулировать и излагать мысли в чёткой логической последовательности, аргументировать и отстаивать своё мнение;
- проявление интереса к исследовательской и проектной деятельности, умение планировать, структурировать и разрабатывать проекты, навыки организации и реализации проектной деятельности.

**II. Комплекс организационно-педагогических условий реализации
общеразвивающей программы**

1. Календарный учебный график на 2024–2025 учебный год

Таблица 2

№ п/п	Основные характеристики образовательного процесса	
1.	Количество учебных недель	36
2.	Количество учебных дней	36
3.	Количество часов в неделю	3
4.	Количество часов	108
5.	Недель в I полугодии	16
6.	Недель во II полугодии	20
7.	Начало занятий	9 сентября
8.	Выходные дни	1 января – 7 января
9.	Окончание учебного года	31 мая

2. Условия реализации программы

Материально-техническое обеспечение:

Требования к помещению:

- помещение для занятий, отвечающие требованиям СанПин 2.4.3648-20 санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи;
- качественное освещение;
- столы, стулья по количеству обучающихся и 1 рабочее место для педагога.

Оборудование:

- компьютеры с подключенными клавиатурами, мышами, мониторами для преподавателя и обучающихся;
- телевизоры Samsung;
- web-камера;
- Wi-Fi роутер.

Расходные материалы:

- бумага писчая;
- маркеры для белой доски;
- шариковые ручки.

Информационное обеспечение:

Программное обеспечение: IDE Visual Studio с установленными пакетами для работы с языками C/C++, C#, Ubuntu, Unity, браузер Yandex последней версии, МойОфис, операционная система Windows/Linux.

Кадровое обеспечение:

Реализовывать программу могут педагоги дополнительного образования, имеющие высшее образование (бакалавриат), среднее профессиональное образование, владеющие педагогическими методами и приемами, знающие особенности технологии обучения основам программирования хотя бы на одном из следующих языков: C, C++, C#.

3. Формы аттестации и оценочные материалы

Система контроля знаний и умений учащихся представляется в виде учёта индивидуального результата по итогам выполнения практических заданий, отдельных кейсов и посредством наблюдения, отслеживания динамики развития учащегося, по каждому контрольному мероприятию и подведения в итоге суммарного балла для каждого обучающегося.

Система отслеживания результатов обучающихся выстроена следующим образом:

- входная диагностика;
- промежуточный контроль;
- итоговый контроль.

В начале обучения, на первом занятии, проводится входная диагностика. Для проведения входной диагностики используется тест (Приложение 1).

Текущий контроль осуществляется регулярно во время занятий. Контроль теоретических знаний осуществляется с помощью педагогического наблюдения, опросов, решения задач, практических работ. В практической деятельности результативность оценивается качеством выполнения работ учащихся, где анализируются положительные и отрицательные стороны работ, корректируются недостатки.

Промежуточный контроль реализуется посредством оценки решения задач и тестирования (Приложение 2). Максимальное количество баллов, которое возможно получить по результатам промежуточного контроля – 50 баллов.

Для подведения итогов по окончании обучения проводится контрольное мероприятие – защита итогового проекта.

Защита итогового проекта осуществляется путем выступления-презентации обучающимся или командой обучающихся. Презентация должна включать в себя тему проекта, его цели и задачи, результаты, средства, которыми были достигнуты полученные результаты. Презентация может быть

выполнена любым удобным наглядным показательным способом (видеоролик, презентация и т. п.). Бланк оценки итоговых проектов представлен в Приложении 3. Максимальное количество баллов за выполнение итогового проекта – 50 баллов.

Результаты промежуточного контроля и защиты итогового проекта суммируются. Итоговая аттестация учащихся осуществляется по 100-балльной шкале, которая переводится в один из уровней освоения образовательной программы согласно таблице 3:

Таблица 3

Баллы, набранные обучающимися	Уровень освоения
1-39 баллов	Низкий
40-79 баллов	Средний
80-100 баллов	Высокий

Формы проведения итогов по каждой теме и каждому разделу общеразвивающей программы соответствуют целям и задачам ДООП.

3.1. Перечень диагностического материала для осуществления мониторинга личностных и метапредметных планируемых результатов

1. Мониторинг достижения обучающимися метапредметных результатов (Приложение 4);

2. Мониторинг достижения обучающимися личностных результатов (Приложение 5).

4. Методические материалы

Особенности организации образовательного процесса:

образовательный процесс осуществляется в очной форме с применением электронного обучения и дистанционных образовательных технологий.

В образовательном процессе используются следующие ***методы обучения:***

- объяснительно-иллюстративный;
- метод проблемного изложения (постановка проблемы и решение её самостоятельно или группой);
- проектно-исследовательский;
- наглядный (демонстрация плакатов, схем, таблиц, диаграмм; использование технических средств; просмотр видеоматериалов);
- практический (практические задания; анализ и решение проблемных ситуаций и т. д.).

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

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

Формы организации образовательного процесса: индивидуальная; групповая.

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 детьми образовательной программы, в соответствии с возрастом обучающихся, составом группы, содержанием учебного модуля: беседа, лекция, мастер-класс, практическое занятие, защита проектов, соревнование.

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 преподавателем с учётом конкретных задач, варианты демонстрационных программ, материалы по терминологии ПО, учебная литература.

Список литературы

Список литературы, использованной при написании программы:

1. Авакян Н.А., Тимофеева Е.В. Информатика: Наглядный школьный курс: удобно и понятно. Тимофеева Е.В. М.: Эксмо-Пресс, 2022 – 176 с.;
2. Марапулец Ю.В. Язык С++. Основы программирования. Марапулец Ю.В. Петропавловск- Камчатский: КамГУ им. Витуса Беринга, 2019. – 158 с.;
3. Сикорд Роберт С. Эффективный С. Профессиональное программирование. Роберт Сикорд – СПб.: Питер, 2022. – 304 с.;
4. Спрингер Вильям. Гид по Computer Science. Вильям Спрингер – СПб.: Питер, 2021. – 304 с.;
5. Стиллмен Э., Грин Д. Head First. Изучаем С#. Эндрю Стиллмен, Дженнифер Грин – СПб.: Питер, 2022. – 768 с.;
6. Таненбаум Эндрю. Архитектура компьютера. Таненбаум Эндрю, Остин Годд. – 6-е изд. - СПб.: Питер, 2022–816 с.

Электронные ресурсы:

1. Code Basics: обучение базовым аспектам языков программирования от образовательной платформы Hexlet. // [Электронный ресурс] - URL: <https://ru.code-basics.com/> (дата обращения: 20.04.2024);
2. METANIT: сайт о программировании. // [Электронный ресурс] - URL: <https://metanit.com/> (дата обращения: 22.04.2024).

Список литературы для обучающихся:

1. Таненбаум Э., Бос Х. Современные операционные системы. Таненбаум Эндрю, Бос Херберт. 4-е изд. – СПб.: Питер, 2022–1120 с.;
2. Страуструп Б. Язык программирования С++. Краткий курс, Бьярне Страуструп – СПб.: ООО «Диалектика», 2019–320 с.

Входная диагностика
(максимум 10 баллов)

ФИО _____ Группа _____

1. К расширениям аудио-файлов относятся: (1 балл)

- 1) exe, txt, msi
- 2) flac, aac, mpa
- 3) iso, odt, ai

2. Опишите своими словами, что такое язык программирования: (2 балла)

3. Сколько килобайт в гигабайте? (1 балл)

4. К архитектурам процессора относятся: (2 балла)1) X86

- 1) RISC
- 2) .NET
- 3) AMD

5. Среда разработки программного обеспечения? (2 балл)

6. Соотношение пикселей по сторонам 4K-разрешения: (1 балла)1) 1440 x 3440

- 1) 3960 x 2080
- 2) 2048 x 1080

7. Элемент компьютера, способный хранить информацию только при включенном компьютере? (1 балл)

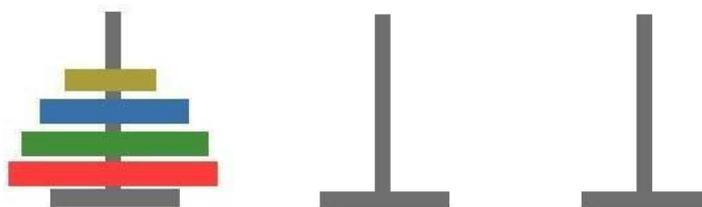
- 1) Процессор
- 2) Оперативная память
- 3) Жесткий диск
- 4) SSD

Пример промежуточного контроля

(максимум – 50 баллов)

Задача: Ханойская башня (25 баллов)

Есть три стержня. На одном из которых нанизаны диски. Диски располагаются в виде пирамидки (ханойской башни): в самом низу лежит самый большой диск, затем идёт чуть поменьше диск, затем ещё меньше диск и т. д. Необходимо переместить диски с одного стержня на другой. Можно использовать все три стержня, но при условии: перекладывать можно только по одному диску за ход, складывать диски можно только меньший на больший.



Тестирование:

1. Что будет в выводе данного кода? (3 балла)

```
#include <iostream>

void print(char const *str) { std::cout << str; }
void print(short num) { std::cout << num; }

int
main()
{ print(
“abc”);
print(0);
print(‘A
’); }
```

- 1) 36978
- 2) Ошибка компиляции

3) 0

4) 1

2. Опишите своими словами, чем интерпретируемые языки отличаются откомпилируемых? (3 балла)

3. Что из следующего автоматически добавляется к каждому классу, если мы не напишем свой собственный при использовании языка C++? (2 балла)

- 1) copy constructor
- 2) оператор присваивания
- 3) Конструктор без параметров
- 4) Все вышеперечисленное

4. Язык программирования C++: (2 балла)

- 1) Строго типизированный и интерпретируемый
- 2) Динамически типизированный и компилируемый
- 3) Строго типизированный и компилируемый
- 4) Динамически типизированный и интерпретируемый

5. Опишите своими словами сферы применения C-подобных языков программирования (5 баллов)

6. Опишите своими словами, чем отличаются высокоуровневые языки программирования от низкоуровневых? Приведите примеры (10 баллов)

Бланк оценки итоговых проектов

(Максимум – 50 баллов)

№ п/п	ФИ автора (ов)	Название проекта	Время защиты	Критерий 1 Актуальность проекта (0-10 б)	Критерий 2 Используемые инструменты (0-10 б)	Критерий 3 Практическая реализация, визуальная составляющая (0-10 б)	Критерий 4 Качество кода/ визуальная составляющая (0-10 б)	Критерий 5 Защита проекта (представление работы) (0-10 б)	Итого

_____ / _____

подпись

расшифровка

Мониторинг достижения обучающимися личностных результатов

за 20_20_ учебный год

№ п/п	Ф.И. обучающегося	возраст	проявление аккуратности при работе с компьютерным оборудованием			сформированы коммуникативные компетентности в общении и сотрудничестве со сверстниками в процессе образовательной, учебно- исследовательской и проектной деятельности			проявление ответственного отношения к обучению, упорства в достижении результата		
			входящий	промежуточный	итоговый	входящий	промежуточный	итоговый	входящий	промежуточный	итоговый
1											
2											
3											
4											
5											
6											
7											
8											
9											
10											

3 балла – качество проявляется систематически
 2 балла – качество проявляется ситуативно
 1 балл – качество не проявляется

Значение показателя по группе:
 1–1.7 – низкий уровень развития качества в группе
 1.8–2.5 – средний уровень развития качества в группе
 2.6–3 – высокий уровень развития качества в группе

**Мониторинг достижения обучающимися метапредметных результатов
за 20_20_ учебный год**

№ п/п	Ф.И. обучающегося	возраст	проявление умения искать, извлекать и отбирать нужную информацию из открытых источников			проявление умения формулировать и излагать мысли в чёткой логической последовательности, аргументировать и отстаивать своё мнение			проявление интереса к исследовательской и проектной деятельности, умение планировать, структурировать и разрабатывать проекты, навыки организации и реализации проектной деятельности		
			входящий	промежуточн ый	итоговый	входящий	промежуточн ый	итоговый	входящий	промежуточн ый	итоговый
1											
2											
3											
4											
5											
6											
7											
8											
9											
10											

3 балла – качество проявляется систематически
2 балла – качество проявляется ситуативно
1 балл – качество не проявляется

Значение показателя по группе:
1–1.7 – низкий уровень развития качества в группе
1.8–2.5 – средний уровень развития качества в группе
2.6–3 – высокий уровень развития качества в группе

Аннотация

Программа «Базовые навыки программирования на C-подобных языках» имеет техническую направленность, в ходе обучения, обучающиеся приобщаются к инженерно-техническим знаниям в области инновационных технологий, формируют техническое мышление. Программа рассчитана на обучающихся 12–17 лет.

В процессе изучения языка программирования C-подобных языков и объектно-ориентированной парадигмы программирования, обучающиеся разрабатывают несколько разноплановых проектов, требующих от подростков использование разных подходов к проектированию, планированию и аналитике, работы с информацией, электроникой и инструментами смежных областей. Таким образом, у подростков развиваются научно-исследовательские, технические и гуманитарные компетенции.