**Муниципальное бюджетное учреждение дополнительного образования**

**«Полазненский центр дополнительного образования детей**

**«Школа технического резерва»**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | |  | УТВЕРЖДАЮ: | | | | |
| Принята на заседании | | | | | | |  | Директор МБУДО «ПЦДОДД «ШТР» | | | | |
| педагогического совета | | | | | | |  |  | | | | |
|  | | | | | | |  |  | | | | А.Ф. Шулятьев |
| от | « |  | » |  | | 2023 г. |  | « |  | » |  | 2023 г. |
| Протокол № | | | |  |  | |  |  |  |  |  |  |

**Дополнительная общеобразовательная**

**общеразвивающая программа технической**

**направленности**

**«Программирование» на 3 года обучения**

Возраст обучающихся: 11-17 лет

Срок реализации: 3 года обучения

Автор-составитель:

Бажина Наталья Николаевна

педагог дополнительного образования

Полазна

2023

**Оглавление**

[Введение 3](#_Toc81932258)

[1. Пояснительная записка 4](#_Toc81932259)

[1.1 Актуальность и необходимость разработки программы 4](#_Toc81932260)

[1.2 Цель и задачи программы 4](#_Toc81932261)

[1.3 Ожидаемые результаты 5](#_Toc81932262)

[2. Организация образовательного процесса 5](#_Toc81932263)

[2.1 Учет особенностей контингента 5](#_Toc81932264)

[2.2 Особенности реализации программы 5](#_Toc81932265)

[2.3 Критерии оценки эффективности реализации программы 5](#_Toc81932266)

[4. Календарно-тематическое планирование 7](#_Toc81932267)

[5. Содержание программы 10](#_Toc81932268)

[Источники информации 16](#_Toc81932269)

# **Введение**

Мы живём в мире техники, и лишь немногие задумываются, что вся техника содержит в себе программы, написанные людьми с помощью языков программирования.

Программирование - процесс создания компьютерных программ. Но программирование - это не просто процесс, это искусство создания программ. Так же, как и скульптор вырезает свою скульптуру, художник пишет свою картину, так и программист воодушевленный своей идеей постепенно создает свою совершенную программу. Ведь если подумать, то весь виртуальный мир существуют только благодаря программированию и людям, которые были вдохновлены своими идеями.

Один из богатейших людей планеты, изобретатель, программист и бизнесмен Билл Гейтс навсегда вошел в историю компьютерных технологий как основатель и крупнейший акционер компании Microsoft. В тринадцать лет Билл написал свою первую программу - игру «Крестики-нолики» на языке программирования BASIC.

Соучредитель компании Apple, разработчик компьютеров американец Стив Возняк по праву считается одним из отцов революции персональных компьютеров в 1970-х гг. Возняк основал Apple Computer вместе со Стивом Джобсом в 1976 году и разрабатывал для первых компьютеров как программную, так и аппаратную часть.

В ряду с известными программистами и хакерами - Линус Торвальдс. Линус создал Linux - ядро операционной системы GNU/Linux, являющейся на данный момент самой распространённой из свободных операционных систем. На ядре Linux построена операционная система Android, являющаяся самой распространенной ОС для смартфонов в мире.

Человек 2010 года по версии журнала Time, самый молодой миллиардер Марк Цукерберг - создатель всемирно известной и крайне популярной социальной сети Facebook. В школьные годы Марк разработал сетевую версию игры "Риск". Отвергнув предложения о трудоустройстве со стороны AOL и Microsoft, совместно с Крисом Хьюзом и Дастином Московицем начал работу над крупнейшей социальной сетью в мире.

Любые компьютерные программы, простой калькулятор или операционная система, социальные сети или форумы, все это держится на языках программирования. Компьютеры присутствуют везде: в экономике, от сферы производства до сферы торговли. Без них и без тех программ, на которых они работают, было бы сложно себе представить, как все бы работало.

# **1. Пояснительная записка**

## 

## **1.1 Актуальность и необходимость разработки программы**

Школьная программа обучения зачастую предоставляет лишь набор начальных знаний и базовых понятий использования компьютера, оставляя не раскрытым истинный потенциал ребенка. 

Программирование – это не просто знание машинного языка, это способность сформулировать проблему и решить ее путем грамотного определения алгоритма действий, четкой систематизации имеющихся знаний и умения применить их на практике. Продолжительная работа с компьютером дает ребенку понимание логики и основных принципов построения и функционирования компьютерных систем. В будущем ему будет проще научиться работать с любой компьютерной программой, действуя интуитивно.

Изучение программирования открывает детям новые возможности и инструменты для самовыражения самыми невероятными способами:

* Управлять роботами и машинами,
* Перекладывать на компьютер решение сложных задач,
* Превращать идеи в виртуальную реальность,
* Делиться идеями с миллионами других.

Изучение программирование не сложнее обучения иностранным языкам. Игры и приложения, использующие визуальные языки программирования, обучают логике и концепциям программирования еще до того, как ребенок научится читать. Изучая программирование, ребенок начинает увлекаться разработкой программ. Простые идеи будут, ведет к придумыванию новых технологий и способов самовыражения.

*Обучение написанию программ расширяет ваш кругозор, помогает лучше думать и создает способы мышления, ориентирующие вас на вещи, которые, я думаю, помогают во всех областях.*

*Билл Гейтс.*

В ходе следующей десятилетки программист будет одной из самых востребованных профессий.

## **1.2 Цель и задачи программы**

*Цель образовательной программы* – развитие у ребёнка интереса, желания и умения преодоления трудностей технологического мира для достижения финансового и нематериального успеха, самореализации в технической сфере общества и рынка путём моделирования различных задач и проблем при программировании различных приложений.

*Задачи образовательной программы*:

- Зажечь в детях тягу к сфере информационных технологий, на примере педагогов образовательного учреждения, показать перспективы занятий программированием, его применимость в современном обществе. Заложить основу и направление будущего инженерного развития.

- Предоставить возможность для изучения программирования. Заложить принципы самостоятельного обучения.

- Пройти путь от идеи и проектирования до реализации программы.

- Сформировать навыки работы со сложным оборудованием и программным обеспечением.

- Привить культуру командной работы и в то же время развить лидерские способности в соревнованиях между школьниками.

## **1.3 Ожидаемые результаты**

Развитие познавательных способностей учащихся, формирование у них алгоритмического мышления, получение реального опыта разработки проектов, повышение интереса учащихся к профессиональному программированию.

# **2. Организация образовательного процесса**

Занятия проходят два раза в неделю по два академических часа. Программа реализуется в течение учебного года.

## **2.1 Учет особенностей контингента**

Программа рассчитана на возраст детей от 11 до 17 лет. Поэтому необходимо правильно выставлять приоритеты в реализации программы.

В этом возрасте ребята готовы к решению конкретных практических задач. Но детям от 10 до 12 лет рекомендуется начать с освоения компьютерной графики и постепенно перейти к программированию. Детям от 11 до 17 лет можно сразу начинать обучение с программирования.

## 

## **2.2 Особенности реализации программы**

Специфика предмета позволяет проводить занятия в различной форме. Могут быть использованы интерактивные игры, требующие технического творчества, соревнования для развития лидерских и командных навыков, научно-практические семинары могут быть полезны для привития ответственности и опыта выступления перед публикой. Данная программа позволяет использовать все перечисленные подходы в обучении детей.

Также следует учитывать, что прогресс не стоит на месте, особенно это касается информационных технологий и микроэлектроники. Поэтому педагог обязан постоянно следить за инновациями в области программирования и стараться максимально внедрить новые подходы во время реализации программы. Так, например, обновление программного обеспечения позволит использовать самые передовые методики написания программного кода.

## **2.3 Критерии оценки эффективности реализации программы**

Каждое занятие состоит из изучения нового материала, работы над созданием программы, где ученики применяют полученные знания и небольших соревнований в конце урока. Такое построение урока дает понимание ценности потраченного времени и прививает ответственность за результат. Большие проекты также проверяются в форме соревнований и конференций, но на уровне школы. Поэтому рекомендуется как можно чаще участвовать в городских, краевых и российских соревнованиях. Проводить собственные мероприятия и форумы с привлечением экспертов, специалистов, прессы и высоких чинов нашего государственного аппарата. Чем больше важности будет оказано деятельности учеников, тем выше результаты будут получены в реализации программы. При подведении итогов следует отмечать удовлетворенность учениками от прохождения образовательной программы, активность и инициативность в образовательном процессе, желание публично демонстрировать результат.

# **3. Учебно-методический план**

1-й год обучения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Тема** | **Всего часов, из них** | | **Форма контроля** |
| **Теория** | **Практика** |
| 1 | Вводная информация о Microsoft Visual C# Express Edition. | 1 | 1 | контрольные задания |
| 2 | Выполнение примеров программ. Знакомство с понятиями языка C#. | 1 | 3 | контрольные задания |
| 3 | Классы и объекты в языке, их свойства. | 1 | 1 | контрольные задания |
| 4 | Задачи по теме «Линейные программы». | 1 | 5 | контрольные задания |
| 5 | Задачи по теме «Развилка». | 1 | 5 | контрольные задания |
| 6 | Задачи по теме «Оператор выбора». | 1 | 5 | контрольные задания |
| 7 | Задачи по теме «Циклы». | 1 | 9 | контрольные задания |
| 8 | Задачи по теме «Целочисленная арифметика». | 1 | 7 | контрольные задания |
| 9 | Задачи по теме «Одномерные массивы». | 1 | 7 | контрольные задания |
| 10 | Задачи по теме «Графика». | 1 | 7 | контрольные задания |
| 11 | Создание приложений в WindowsForms. Основные инструменты | 1 | 3 | контрольные задания |
| 12 | Создание полезных приложений в WindowsForms | 6 | 58 | контрольные задания |
| 13 | Компьютерная графика. Gimp. | 1 | 7 | контрольные задания |
|  | **Итого** | **18** | **118** |  |

2-й год обучения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Тема** | **Всего часов, из них** | | **Форма контроля** |
| **Теория** | **Практика** |
| 1 | Создание полезных приложений в WindowsForms.  Повторение пройденного материала. | 2 | 54 | контрольные задания |
| 2 | Разработка игр на Unity 3D | 10 | 50 | контрольные задания |
| 3 | Основы Web-программирования | 10 | 10 | контрольные задания |
|  | **Итого** | **22** | **114** |  |

3-й год обучения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Тема** | **Всего часов, из них** | | **Форма контроля** |
| **Теория** | **Практика** |
| 1 | Создание полезных приложений в WindowsForms | 4 | 30 | контрольные задания |
| 2 | Разработка игр на Unity 3D | 10 | 24 | контрольные задания |
| 3 | Python для начинающих | 34 | 34 | контрольные задания |
|  | **Итого** | **48** | **88** |  |

# **4. Календарно-тематическое планирование**

1-й год обучения

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Тема** | **Содержание** | **Всего часов, из них** | | **Форма контроля** |
| **Теория** | **Практика** |
| 1 | Вводная информация о Microsoft Visual C# Express Edition. Создание новой программы (проекта) в Visual C# Express. | 1.1 Ответ на вопрос: для чего нужно программирование? Рассказ об истории развития языков программирования. | 1 | 1 | контрольные задания |
| 1.2 Знакомство с C#. Основные отличия от прочих языков. |
| 1.3. Знакомство со средой программирования Microsoft Visual C# Express Edition. Описание интерфейса, создание нового проекта консольной программы. |
| 2 | Выполнение примеров программ. Знакомство с понятиями языка C#. | 2.1. Основные понятия языка программирования. | 1 | 3 | контрольные задания |
| 2.2. Рассказ о возможностях программирования в среде разработки Microsoft Visual C# Express Edition. |
| 2.3. Выполнение базовых примеров. |
| 3 | Классы и объекты в языке, их свойства. Создание объектов. Типы «String» (строка), числовые, «Boolean» (логическое значение) | 3.1. Рассказ о классах и объектах. Проведение параллелей с реальным миром. | 1 | 1 | контрольные задания |
| 3.2. Типы “string”, “int”, “double” и “boolean”. Рассказ о том для чего они служат и как их применять. |
| 3.3. Применение типов и классов на примере простых консольных приложений. |
| 4 | Задачи по теме «Линейные программы». | 4.1. Вычисления по формулам. | 1 | 5 | контрольные задания |
| 4.2. Вычисления в математических задачах. |
| 4.3. Задачи на составление логических выражений. |
| 4.4. Области, описываемые логическими выражениями. |
| 5 | Задачи по теме «Развилка». | 5.1. Текстовые задачи. | 1 | 5 | контрольные задания |
| 5.2. Вычисление значений функции. |
| 6 | Задачи по теме «Оператор выбора». | 6.1. Решение задач с использованием оператора выбора. | 1 | 5 | контрольные задания |
| 7 | Задачи по теме «Циклы». | 7.1. Циклы с параметром. | 1 | 9 | контрольные задания |
| 7.2. Ряды |
| 7.3. Вычисление последовательностей. |
| 7.4. Табулированные функции. |
| 8 | Задачи по теме «Целочисленная арифметика». | 8.1. Решение задач по теме «Целочисленная арифметика». | 1 | 7 | контрольные задания |
| 9 | Задачи по теме «Одномерные массивы». | 9.1. Решение задач с использованием одномерных массивов. | 1 | 7 | контрольные задания |
| 9.2. Сортировка одномерного массива. |
| 10 | Задачи по теме «Графика». | 10.1. Задачи по теме «Графика». | 1 | 7 | контрольные задания |
| 11 | Создание приложений в WindowsForms. Основные инструменты | 11.1. Создание нового проекта (приложения для windows) в среде разработки. Описание интерфейса по созданию приложений. | 1 | 3 | контрольные задания |
| 11.2. Разработка программы с кнопкой в центре окна. |
| 12 | Создание полезных приложений в WindowsForms | 12.1. Создание полезных приложений в WindowsForms | 6 | 58 | контрольные задания |
| 13 | Компьютерная графика. Gimp. Оновные инструменты. Создание простейшей gif анимации | 13.1. Знакомство с 2D графикой. Основные инструменты | 1 | 7 | контрольные задания |
| 13.2. Простейшая анимация в Gimp. Создание слоев |
|  | **Итого** |  | **18** | **118** |  |

2-й год обучения

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Тема** | **Содержание** | **Всего часов, из них** | | **Форма контроля** |
| **Теория** | **Практика** |
| 1 | Создание полезных приложений в WindowsForms.  Повторение пройденного материала. | 1.1 Создание полезных приложений в WindowsForms | 2 | 54 | контрольные задания |
| 2 | Разработка игр на  Unity 3D | 2.1 Введение в Unity  2.2 Физика и префабы в Unity  2.3 Создание окружения  2.4 Основы программирования. Типы данных и методы  2.5 Основы программирования. Классы и объекты  2.6 Программирование и управление персонажем  2.7 Свет и звук  2.8 Анимация  2.9 Пользовательский интерфейс  2.10 Поиск пути в Unity | 10 | 50 | контрольные задания |
| 3 | Основы Web-программирования | 3.1 «Среда и стандарты Инернет-разработок»  3.2 «Стилевое и скриптовое воздействие на документ»  3.3 «Функциональность серверных ASP-сценариев» | 10 | 10 | контрольные задания |
|  | **Итого** |  | **22** | **114** |  |

3-й год обучения

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Тема** | **Содержание** | **Всего часов, из них** | | **Форма контроля** |
| **Теория** | **Практика** |
| 1 | Создание полезных приложений в WindowsForms | Создание полезных приложений в WindowsForms | 4 | 30 | контрольные задания |
| 2 | Разработка игр на Unity 3D | Разработка игр на Unity 3D | 10 | 24 | контрольные задания |
| 3 | Python для начинающих | Знакомство с Python.  Команды input() и print().  Параметры sep, end.  Переменные.  Работа с целыми числами  Условный оператор.  Логические операции.  Вложенный и каскадный условный оператор.  Типы данных Встроенные функции.  Цикл for.  Функция range().  Цикл с предусловием.  Операторы break, continue, else.  Вложенные циклы.  Строковый тип данных: индексация и срезы  Методы строк  Резервное время.  Введение в списки.  Основы работы со списками.  Вывод элементов списка.  Методы списков.  Функции  Локальные и глобальные переменные.  Функции возвращающие значения.  Работа над проектом | 34 | 34 | контрольные задания |
|  | **Итого** |  | **48** | **88** |  |

# **5. Содержание программы**

1-й год обучения

**Тема 1. Вводная информация о Microsoft Visual C# Express Edition. Создание новой программы (проекта) в Visual C# Express.**

*1.1. Ответ на вопрос: для чего нужно программирование? Рассказ об истории развития языков программирования.*

Положение программирования в современном мире. Компьютер, как инструмент для созидания. Сравнение реального мира и программ, выявление сходств, моделирование событий и объектов на языке программирования. Основные вехи в разработке приложений. История развития языков программирования. Современные технологии: программные средства и их возможности.

*1.2. Знакомство с C#. Основные отличия от прочих языков.*

Позиционирование C#, как самого современного, мощного и удобного языка программирования. Кратко о компании Microsoft и причинах, побудивших их создать этот язык. Основные свойства языка. Описание типов программ, которые мы можем создавать на C#. Основные отличия от конкурентов.

*1.3. Знакомство со средой программирования Microsoft Visual C# Express Edition. Описание интерфейса, создание нового проекта консольной программы.*

Способы создания приложений на C# (от блокнота до MS Visual Studio 2010). Знакомство с бесплатной средой программирования Microsoft Visual C# Express Edition. Где можно ее загрузить. Инструкции по установке. Описание интерфейса, основных действий, этапов создания проекта, способов поиска информации и справки.

**Тема 2. Выполнение примеров программ. Знакомство с понятиями языка C#.**

*2.1. Основные понятия языка программирования.*

Рассказ об основных понятиях языка программирования на примере общения с неким пришельцем, ничего не знающем ничего о нашей планете. Воплотить попытку объяснить что-либо ему, тем самым приблизится к пониманию таких понятий как «Класс», «Тип», «Поле», «Событие», «Процедура» и «Функция».

*2.2. Рассказ о возможностях программирования в среде разработки Microsoft Visual C# Express Edition.* Встроенная поддержка для рефакторинга программного кода. Расширение и завертывание кода в определенный контекст. Интегрированная система документации. Встроенная справка и помощь в Visual C# Express Edition.

*2.3. Выполнение базовых примеров.*

Необходимо показать ученикам близкую связь между реальным миром и смоделированными в программе событиями и объектами. На простых примерах, не вдаваясь в детали, ученики увидят, что программы умеют многое. Цель – зародить интерес к программированию.

**Тема 3. Классы и объекты в языке, их свойства. Создание объектов. Типы «String» (строка), числовые, «Boolean» (логическое значение)**

*3.1. Рассказ о классах и объектах. Проведение параллелей с реальным миром.*

Связь программы с реальным миром. Разделение реальных объектов на классы и подбор программной модели выбранных классов. Так же для классов определяем некоторые обобщенные свойства, называемые полями. Проходим материал на примере роботов и людей, далее каждый сам разрабатывает свои классы.

*3.2. Типы “string”, “int”, “double” и “boolean”. Рассказ о том для чего они служат и как их применять.*

Рассказ о том для чего они служат и как их применять. Подробное описание основных типов. Пытаемся продумать какие свойства объектов реального мира можем описать этими 4мя типами данных. Применяем их для наших классов робот и человек.

*3.3. Применение типов и классов на примере простых консольных приложений.*

Вместе создаем приложение, в котором обращаем внимание на классы и их поля. Создаем классы на две группы объектов реального мира: роботы и люди.

**Тема 4. Задачи по теме «Линейные программы».**

*4.1. Вычисления по формулам.*

Вычислить значение выражения по формуле (все переменные принимают действительные значения). Пример типовых задач:

* ;
* ;
* .

*4.2. Вычисления в математических задачах.*

Пример типовых задач:

* Вычислить периметр и площадь прямоугольного треугольника по длинам а и b двух катетов.
* Вычислить длину окружности и площадь круга одного и того же заданного радиуса R.
* Даны два числа. Найти среднее арифметическое кубов этих чисел и среднее геометрическое модулей этих чисел.

*4.3. Задачи на составление логических выражений.*

Составить линейную программу, печатающую true, если указанное высказывание является истинным, и false в противном случае. Пример типовых задач:

* Сумма двух первых цифр заданного четырехзначного числа равна сумме двух его последних цифр.
* Точка с координатами (х, у) принадлежит части плоскости, лежащей между прямыми х— т, х= п (т<п).
* Целое число N является четным двузначным числом.

*4.4. Области, описываемые логическими выражениями.*

Для данных областей составить линейную программу, которая печатает true, если точка с координатами (х, у) принадлежит закрашенной области, и false в противном случае.

**Тема 5. Задачи по теме «Развилка».**

*5.1. Текстовые задачи.*

Пример типовых задач:

* Даны три действительных числа. Возвести в квадрат те из них, значения которых неотрицательны, и в четвертую степень — отрицательные.
* Даны два угла треугольника (в градусах). Определить, существует ли такой треугольник, и если да, то будет ли он прямоугольным.
* На плоскости задана своими координатами точка А. Указать, где она расположена (на какой оси или в каком координатном угле).

*5.2. Вычисление значений функции.*

Пример типовых задач:

  

**Тема 6. Задачи по теме «Оператор выбора».**

Пример типовых задач:

* Написать программу, которая по номеру дня недели (целому числу от 1 до 7) выдает в качестве результата количество уроков в вашем классе в этот день.
* Составить программу, которая по заданным году и номеру месяца, определяет количество дней в этом месяце.
* Составить программу, которая по данному числу (1—12) выводит название соответствующего ему месяца.

**Тема 7. Задачи по теме «Циклы».**

*7.1. Циклы с параметром.*

Пример типовых задач:

* Начав тренировки, спортсмен в первый день пробежал 10 км. Каждый день он увеличивал дневную норму на 10% нормы предыдущего дня. Какой суммарный путь пробежит спортсмен за 7 дней?
* Одноклеточная амеба каждые 3 часа делится на 2 клетки. Определить, сколько амеб будет через 3, 6, 9, 12,..., 24 часа.
* У гусей и кроликов вместе 64 лапы. Сколько может быть кроликов и гусей (указать все сочетания)?

*7.2. Ряды*

Пример типовых задач:

* Дано натуральное число N. Вычислить

.

* Дано натуральное число N. Вычислить произведение первых N сомножителей

.

* Дано действительное число x. Вычислить

.

*7.3. Вычисление последовательностей.*

Даны числовой ряд и некоторое число “е”. Найти сумму тех членов ряда, модуль которых больше или равен заданному е. Общий член ряда имеет вид:

* ;
* ;
* .

*7.4. Табулированные функции.*

Составить программу для вычисления значений функции F(x) на отрезке [а, b] с шагом h. Результат представить в виде таблицы, первый столбец которой — значения аргумента, второй - соответствующие значения функции.

* ;
* ;
* .

**Тема 8. Задачи по теме «Целочисленная арифметика».**

Пример типовых задач:

* Дано натуральное число n. Найти сумму первой и последней цифры этого числа.
* Даны два натуральных числа m и n (m< 9999, n < 9999). Проверить, есть ли в записи числа т цифры, совпадающие с цифрами в записи числа n.
* Дано натуральное число n < 99. Дописать к нему цифру k в конец и в начало.

**Тема 9. Задачи по теме «Одномерные массивы».**

*9.1. Решение задач с использованием одномерных массивов.*

Пример типовых задач:

* В целочисленной последовательности есть нулевые элементы. Создать массив из номеров этих элементов.
* Дана последовательность действительных чисел. Выяснить, будет ли она возрастающей.
* Дана последовательность. Указать наименьшую длину числовой оси, содержащую все эти числа.

*9.2. Сортировка одномерного массива.*

Пример типовых задач:

* Заданы два одномерных массива с различным количеством элементов и натуральное число k. объединить их в один массив, включив второй массив между k-м и (k+1)-м элементами первого, при этом, не используя дополнительный массив.
* Сортировка выбором. Дана последовательность чисел а1, а2, ..., аn. Требуется переставить элементы так, чтобы они были расположены по убыванию. Для этого в массиве, начиная с первого, выбирается наибольший элемент и ставится на первое место, а первый — на место наибольшего. Затем, начиная со второго, эта процедура повторяется. Написать алгоритм сортировки выбором.
* Сортировка обменами. Дана последовательность чисел а1, а2, ..., аn. Требуется переставить числа в порядке возрастания. Для этого сравниваются два соседних числа ai и ai+1. Если ai > ai+1 то делается перестановка. Так продолжается до тех пор, пока все элементы не станут расположены в порядке возрастания. Составить алгоритм сортировки, подсчитывая при этом количества перестановок.

**Тема 10. Задачи по теме «Графика».**

Построить чертежи к следующим типовым задачам:

* В треугольной пирамиде построить сечение, параллельное основанию.
* В треугольной пирамиде построить сечение, проходящее через среднюю линию боковой грани и противоположную вершину основания.
* В правильной четырехугольной пирамиде провести сечение, параллельное основанию и проходящее через середину бокового ребра.

**Тема 11. Создание приложений в WindowsForms. Основные инструменты**

*11.1. Создание нового проекта (приложения для windows) в среде разработки. Описание интерфейса по созданию приложений.*

Подробное описание возможностей среды разработки по созданию приложений для Windows. Пояснения для чего необходимы приложения. Вся группа создает новый проект, следуя инструкциям преподавателя.

*11.2. Разработка программы с кнопкой в центре окна.*

Необходимо, что бы каждый смог реализовать простейшее приложение Windows – форму с единственной кнопкой на нем

**Тема 12. Создание полезных приложений в WindowsForms**

Создание полезных приложений: калькулятор, блокнот, часы с таймером, простые игрушки на логику и т.д.

**Тема 13. Компьютерная графика. Gimp. Оновные инструменты. Создание простейшей gif анимации**

*13.1. Знакомство с 2D графикой. Основные инструменты*

Отличие растровой от векторной графики. Gimp как простейший графический редактор и инструмент для создания анимации. Работа с кистями, преобразование объектов и т.д.

*13.2. Простейшая анимация в Gimp. Создание слоев*

Работа со слоями, создание каждого кадра анимации и преобразование в gif анимацию

2-й год обучения

**Тема 1. Создание полезных приложений в WindowsForms.**

Создание полезных приложений. «Математическая головоломка», «Найди пару». Работа с часами. Создание теста.

**Тема 2. Разработка игр на Unity 3D**

*2.1 Введение в Unity*

Учащиеся узнают, как устроен редактор Unity и какими функциями обладает. Увидят, что такое окна Scene, Game, Project, Hierarchy, Inspector и Console и научатся сохранять расстановку окон. Поймут, что такое префаб и смогут добавить компоненты на объект.

*2.2 Физика и префабы в Unity*

Учащиеся узнают для чего используются компоненты rigidbody и collider и смогут их настроить. Научатся создавать префабы, варианты префабов и вложенные префабы.

*2.3 Создание окружения*

Учащиеся смогут создавать красивые и реалистичные окружения в играх. Получат практические навыки работы с terrain.

*2.4 Основы программирования. Типы данных и методы*

Учащиеся освоят азы программирования: смогут запрограммировать простые математические задачи, используя переменные, арифметические действия и группируя это в методы.

*2.5 Основы программирования. Классы и объекты*

Учащиеся научатся создавать классы и наследовать их. Узнают, чем отличается класс от объекта, и научатся создавать основные конструкции класса.

*2.6 Программирование и управление персонажем*

Учащиеся научатся настраивать управление в игре от первого лица. Изучат основы геймдизайна. Испытают функционал наследования объектов, доступный в иерархии Unity.

*2.7 Свет и звук*

Учащиеся научатся работать с источником света. Смогут расставлять освещение и запекать его на базовом уровне. Научатся работать с источником звуков и приёмником звуков и научатся создавать звуковое сопровождение.

*2.8 Анимация*

Учащиеся будут создавать и запускать простую анимацию, работать с окном Animation через скрипт.

*2.9 Пользовательский интерфейс*

Научатся создавать в игре функциональное меню. Добавят кнопки и зададут им необходимые функции.

*2.10 Поиск пути в Unity*

Узнают, что такое поиск путей, и научатся пользоваться стандартным поиском.

**Тема 3. Основы Web-программирования**

*3.1 «Среда и стандарты Инернет-разработок»*

* Ключевая терминология сферы Интернет-проектов
* Схема организационного взаимодействия участников
* Параметры удобочитаемости и доступности содержимого ресурсов
* Правовые риски (чего делать нельзя или следует избегать):
* Стандарт гипертекстового документа
* Принципы верстки страницы
* Шапка HTML страницы (<HEAD>)
* Тело HTML страницы (<BODY>)
* Ключевые теги разметки (<BODY>)
* Структурное форматирование текста
* Форматирование гипертекстовых ссылок
* Маркеры элементов списка
* Внедрение иллюстраций тегом <IMG>

*3.2 «Стилевое и скриптовое воздействие на документ»*

* Необходимость стилизации контента
* Режимы подключения стилей
* Правила записи команд стилизации
* К чему применяется стилевое форматирование
* Важнейшие единицы измерения в CSS
* Способы описания цветов в CSS
* Команды стилевого форматирования
* Обобщенная модель отступов
* CSS управление текстурой
* Стилизация поведения ссылок
* Возможное расположение скриптов в структуре страницы
* Синтаксис команд внедрения
* Способы назначения обработчиков событий
* Общие правила языка JavaScript
* Скриптовые библиотеки

*3.3 «Функциональность серверных ASP-сценариев»*

* Исполняемые форматы
* Методы передачи данных (из форм)
* Общие технологические принципы
* Объявление констант
* Объявление переменных
* Разновидности присваивания
* Проверка логического условия
* Множественный выбор
* Циклы
* Специальные типы данных
* Массивы
* Передача параметров сценарию

3-й год обучения

**Тема 1. Создание полезных приложений в WindowsForms.**

Создание полезных приложений.

**Тема 2. Разработка игр на Unity 3D**

Разработка игр на Unity 3D.

**Тема 3. Python для начинающих**

3.1 Знакомство с Python. Команды input() и print().

3.2 Параметры sep, end. Переменные. Комментарии. PEP 8

3.3 Работа с целыми числами

3.4 Условный оператор. Логические операции and, or, not

3.5 Вложенный и каскадный условный оператор

3.6 Типы данных int, float, str. Встроенные функции min(), max(), abs(). Оператор in.

3.7 Цикл for. Функция range().

3.8 Частые сценарии при написании циклов. Расширенные операторы присваивания.

3.9 Цикл с предусловием while

3.10 Операторы break, continue, else.

3.11 Вложенные циклы

3.12 Строковый тип данных: индексация и срезы

3.13 Методы строк

3.14 Резервное время. Введение в списки.

3.15 Основы работы со списками. Методы списков

3.16 Вывод элементов списка. Строковые методы split() и join()

3.17 Методы списков. Списочные выражения

3.18 Функции

3.19 Локальные и глобальные переменные. Функции возвращающие значения.

3.20 Функции возвращающие значения.

3.21 Работа над проектом

# **Источники информации**

1. Семакин И. Г., Шестаков А. П. Основы программирования: Учебник. - М.: Мастерство, 2016.
2. Дрейер М. C# для школьников: Учебное пособие / М. Дрейер. Перевод с англ. под ред. В. Биллига - М.: Интернет-Университет Информационных Технологий; БИНОМ.
3. Троелсен, Эндрю. Язык программирования С# 2008 и платформа .NET 3.5, 4-е изд.: Пер. с англ. - М.: ООО "И.Д. Вильяме", 2018.
4. Гуриков С. Р. Введение в программирование на языке Visual C#; Форум, Инфра-М, 2018. - 448 c.
5. Мартин Р. С., Мартин М. Принципы, паттерны и методики гибкой разработки на языке C#; Символ-Плюс, 2019. - 768 c.
6. Пугачев С., Шериев А., Кичинский К. Разработка приложений для Windows 8 на языке C#; БХВ-Петербург, 2018. - 416 c.
7. Фленов Михаил Библия C#; БХВ-Петербург, 2020. - 560 c.
8. Учебник. Создание средства просмотра изображений приложения Windows Forms в Visual Studio, 2022
9. С. Тимофеев, Работа в графическом редакторе GIMP, 2018 - 351 с.
10. HTML и CSS. Разработка и создание веб-сайтов, Издательство [Эксмо](https://www.bookvoed.ru/books?publisher=36972), серия Мировой компьютерный бестселлер, [2019](https://www.bookvoed.ru/books?yearMin=2019&yearMax=2019) г., - 480 страниц,
11. Васильев, А. Н. Python на примерах - Наука и Техника, 2019 - 432 с.
12. Прохоренок, Н. А. Python 3: самое необходимое - Петербург, 2019 - 608 с.
13. Гэддис, Т. Начинаем программировать на Python - Петербург, 2019 - 768 с.
14. Седжвик, Р. Программирование на языке Python - Вильямс, 2017 - 736 с.
15. Харрисон, М. Как устроен Python. - Питер, 2002 - 272 с.

**Муниципальное бюджетное учреждение дополнительного образования**

**«Полазненский центр дополнительного образования детей**

**«Школа технического резерва»**

**Аннотация Дополнительной общеобразовательной**

**общеразвивающей программы технической направленности**

**«Программирование»**

|  |  |  |
| --- | --- | --- |
| 1. | **Автор программы (ФИО, место работы, должность)** | Бажина Наталья Николаевна,  место работы: МБУДО «ПЦДОД «ШТР»,  педагог дополнительного образования |
| 2. | **Вид программы** | Модифицированная |
| 3. | **Направленность программы** | Научно-техническая |
| 4. | **Дата создания программы** | 2023 год |
| 5. | **Срок реализации программы** | 3 года |
| 6. | **Возраст обучающихся** | 11-17 лет |
| 7. | **Когда и где рассмотрена** | Заседание педагогического совета  Протокол № от 2023 года |
| 8. | **Цель программы** | Развитие у обучающихся интереса к программированию, программному коду; обучение основным принципам алгоритмизации, построения программы и выбора метода выполнения поставленной задачи; формирование технического склада ума в решении информационных задач различной сложности. |
| 9. | **Краткое содержание программы** | - Решение задач по основам алгоритмизации в языках программирования С#, Python.  - Знакомство с современными видами языков программирования, их доли в IT-индустрии.  - Разработка информационных проектов.  - Моделирование и создание продуктивных приложений в Windows Forms.  - Разработка информационных проектов. |
| 10. | **Прогнозируемые результаты** | Учащиеся должны научится составлять алгоритмы решения задач, быстро вводить информацию в код, реализовывать свои идеи в программном виде. |