

Министерство образования и науки Самарской области

Государственное бюджетное образовательное учреждение
дополнительного образования Самарской области
«Самарский областной центр детско-юношеского технического творчества»

Принята на заседании
Методического Совета
Протокол № 2

от « 20 » июня 2023 г.



УТВЕРЖДАЮ
Директор ГБОУ ДО СО СОЦДИЮТТ

/А. Ю. Богатов/
июня 2023 г.

Дополнительная общеобразовательная общеразвивающая
программа технической направленности

«Прикладное программирование»
(IT-квантум, углубленный модуль)

Возраст детей: 11-17 лет
Срок обучения: 1 год

Разработчик:
Сосулин Павел Дмитриевич
педагог дополнительного образования

Тольятти, 2023

ОГЛАВЛЕНИЕ

| | |
|--|----|
| 1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА..... | 3 |
| 2. УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН | 10 |
| 3. СОДЕРЖАНИЕ | 13 |
| 4. РЕСУРСНОЕ ОБЕСПЕЧЕНИЕ | 20 |
| 5. СПИСОК ЛИТЕРАТУРЫ | 24 |
| 6. ПРИЛОЖЕНИЕ 1. КАЛЕНДАРНО-ТЕМАТИЧЕСКИЙ ПЛАН..... | 26 |
| 7. ПРИЛОЖЕНИЕ 2. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ..... | 29 |

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеобразовательная программа «Прикладное программирование» способствует приобщению учащихся к новейшим техническим, информационным технологиям и логическому развитию учащихся посредством творческой и проектной деятельности. Приоритетная задача программы – обучение основам программирования.

Изучая программирование, учащиеся получают глубокое понимание принципов работы компьютера, организации ввода, вывода и хранения информации, принципов построения диалоговых приложений, познают азы профессии программиста.

Современный период развития общества характеризуется масштабными изменениями в окружающем мире, влекущими за собой пересмотр социальных требований к образованию, предполагающими его ориентацию не только на усвоение обучающимся определенной суммы знаний, но и на развитие его личности, а также овладение метапредметными компетенциями. Большими возможностями в развитии школьников обладает подготовка в области информационных технологий и программирования. Программа отвечает потребностям общества, формированию творческих способностей и развитию личности. Этими факторами определяется *выбор уровня и направленности программы.*

Разработка программы опирается на следующие **нормативные документы**:

- Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации» (с изменениями от 29.12.2022 N 642-ФЗ)
- Концепция развития дополнительного образования детей до 2030 года (утверждена распоряжением Правительства РФ от 31.03.2022 № 678-р)
- Приказ Министерства просвещения РФ от 27.07.2022 № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»
- Постановление Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»
- Указ Президента Российской Федерации от 21 июля 2020 года № 474 «О национальных целях развития Российской Федерации на период до 2030 года»
- Стратегия развития воспитания в Российской Федерации на период до 2025 года (утверждена распоряжением Правительства Российской Федерации от 29.05.2015 № 996-р).

Программа имеет **техническую** направленность. Программы технической направленности в системе дополнительного образования ориентированы на развитие технических и творческих способностей и умений учащихся, организацию их научно-исследовательской деятельности и профессионального самоопределения.

Новизна дополнительной общеобразовательной программы «Прикладное программирование» заключается в следующем:

- в основе педагогического подхода лежит вытягивающая модель обучения. Перед обучающимися ставятся задачи, заведомо более сложные, чем те, с которыми они сталкивались в своей практике. Это побуждает к поиску информации, анализу и запросу на получение компетенций, а также формирует самостоятельность и ответственность;
- использование в обучении уникального оборудования даёт возможность реального изготовления спроектированных моделей;
- в рамках программы созданы условия для развития навыков самообразования и исследования, построения индивидуальной траектории обучения, формирования познавательных интересов, интеллектуальной и ценностно-смысловой сферы обучающихся; а также предоставлены возможности участия в конкурсах, выставках и фестивалях различного уровня.

Актуальность программы обусловлена потребностью общества в технически грамотных специалистах в области программирования, а также необходимостью повышения мотивации к выбору инженерных профессий и создания системы непрерывной подготовки будущих квалифицированных инженерных кадров, обладающих профессиональными компетенциями для развития отечественной науки и техники.

Педагогическая целесообразность программы обусловлена творческо-практической направленностью, которая является стратегически важным направлением в развитии и воспитании учащихся. Посредством метода проектной работы и с использованием кейс-технологий в рамках базового модуля обучающиеся углубят свои фундаментальные навыки и базовые знания в сфере IT. Для реализации образовательной программы используются технологии развивающего, исследовательского и проектного обучения, которые обеспечивают выполнение поставленных целей и задач образовательной деятельности.

Технологии развивающего обучения позволяют ориентировать учебный процесс на потенциальные возможности учащихся и их реализацию, вовлекать учащихся в различные виды деятельности.

Исследовательские технологии развивают внутреннюю мотивацию ребёнка к обучению, формируют навыки целеполагания, планирования, самооценивания и самоанализа.

Метод проектов обеспечивает вариативность учебного процесса с учетом уровня подготовки, интересов учащихся и предполагает решение проблемы, предусматривающей, с одной стороны, использование разнообразных методов, средств обучения, а с другой - интегрирование знаний, умений из различных областей науки, техники, технологии, творческих областей.

Отличительные особенности программы.

К основным отличительным особенностям настоящей программы можно отнести следующие:

- преобладающие методы обучения – метод кейсов и метод проектов;
- направленность на формирование ключевых компетенций;
- использование игропрактик;
- создание на занятиях среды для развития умения взаимодействовать в команде;
- направленность на развитие системного мышления.

Целью программы развитие алгоритмического мышления, аналитических и логических компетенций старших школьников и через обучение программированию.

Для успешной реализации поставленной цели необходимо решить следующие **задачи**:

Образовательные:

- получить базовые теоретические знания в области разработки приложений с использованием языка программирования Python и библиотеки PyGame;
- овладеть основами языка программирования Python и современными принципами разработки приложений;
- овладеть приёмами самостоятельной и творческой деятельности при разработке собственных приложений.

Развивающие:

- развивать устойчивый интерес у обучающихся к данной сфере деятельности;
- вовлечь обучающихся в проектную деятельность с формированием опыта деятельности на всех этапах выполнения проекта – от рождения замысла до итогового завершения;
- вовлечь обучающихся в активную познавательную деятельность через индивидуальный проект.

Воспитательные:

- содействовать воспитанию личностных качеств: самостоятельности, уверенности в своих силах, креативности;
- способствовать стимулированию самостоятельности учащихся в изучении теоретического материала и решении практических задач;
- содействовать профессиональному самоопределению обучающихся.

Адресат программы: программа ориентирована на обучение детей 11-16 лет, которые успешно закончили программу вводного модуля IT квантума. Учащиеся этого возраста уже имеют достаточную базовую подготовку для углубленного изучения программирования, участия в конкурсных и соревновательных мероприятиях, поэтому содержание программы адаптировано к данному возрасту.

Наполняемость групп: до 12 человек.

Предполагаемый состав групп: дети возраста 11-16 лет, группа формируется в зависимости от возраста детей.

Условия приёма: в группу принимаются учащиеся, успешно освоившие программу базового модуля IT-квантума, у которых имеется желание дальнейшего развития по этой программе обучения.

Сроки реализации программы: 1 год.

Особенности реализации программы. Количество часов, разделов и тем учебно-тематического плана носят рекомендательный характер. Педагог дополнительного образования может уменьшать или увеличивать количество часов, разделов с учётом интересов, потребностей, уровня подготовки учащихся.

Формы и режим занятий.

Обучение проводится в очной форме. Для усвоения курса и повышения общей эрудированности обучаемых ключевые предметные понятия модуля и значимую фактологическую информацию предлагать в виде дополнительных заданий для самостоятельного выполнения обучающимися, например, с последующим обсуждением решения на учебном занятии или консультации.

Занятия проводятся один раз в неделю. Продолжительность одного учебного занятия 2 академических часа, продолжительность учебного часа – 45 минут. Кратность занятий и их продолжительность обосновывается рекомендуемыми нормами СанПин 2.4.4.3172 – 14 целью и задачами программы.

Программой предусмотрено проведение комбинированных занятий: занятия состоят из теоретической и практической частей, причём большее количество времени занимает именно практическая часть.

Формы занятий. Лекции с выполнением практического задания повторяемого учащимися на аналогичном оборудовании, обсуждения, практические занятия по решению заданий, аналогичных рассмотренным на лекции, метод кейсов и проектов, игровые формы. Также программа курса включает групповые и индивидуальные формы работы обучающихся (в зависимости от темы занятия).

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

Реализация программы предполагает использование здоровьесберегающих технологий: создание безопасных технических условий, благоприятного психологического климата, наличие динамических пауз, периодическая смена деятельности.

Ожидаемые результаты обучения по образовательному компоненту:

- обучающийся знает и эффективно использует язык программирования Python;
- обучающийся понимает принципы и приёмы разработки программ на Python, умеет их отлаживать и запускать на выполнение;
- обучающийся знает ключевые особенности библиотеки Tkinter и умеет их применять при разработке приложений с графическим интерфейсом пользователя на языке Python;
- обучающийся знает ключевые особенности библиотеки PyGame и умеет их применять при разработке игровых приложений на языке программирования Python.

Ожидаемые результаты обучения по развивающему компоненту:

- умеет находить решение проблемы с использованием различных методов генерации идей;
- эффективно использует различные источники информации: интернет, книги и журналы, мнение экспертов;
- умеет самостоятельно планировать пути достижения целей, соотносить свои действия с планируемыми результатами, осуществлять контроль и коррекцию своей деятельности в процессе достижения результата.

Ожидаемые результаты обучения по воспитательному компоненту:

- не нуждается в постоянной помощи педагога, умеет следовать инструкциям;
- умеет работать в группе, доброжелательно и уважительно строить свое общение со сверстниками и взрослыми;
- показывает готовность и способность к саморазвитию и личностному самоопределению;
- демонстрирует высокий уровень мотивации к целенаправленной познавательной деятельности с целью приобретения профессиональных навыков в сфере инженерного творчества;
- соблюдает ТБ, бережно относится к оборудованию и техническим устройствам.

Психолого-педагогический мониторинг результатов образовательного процесса

Психолого-педагогический мониторинг – это систематическая оценка уровня освоения дополнительной программы в течение учебного года. Он складывается из следующих компонентов.

Входной контроль осуществляется на первых занятиях с целью выявления стартового образовательного уровня развития детей в форме анкетирования обучающихся.

Оперативный контроль осуществляется на каждом учебном занятии с целью отслеживания освоения текущего программного материала, коррекции практических умений.

Промежуточный контроль проводится по завершению каждого кейса в форме тестирования или презентации выполненных проектов.

Итоговый контроль выполняется по результатам окончания программы в форме презентации итогового инженерного проекта

В конце учебного года результаты всех диагностических процедур обобщаются и определяется уровень результатов образовательной деятельности каждого обучающегося – интегрированный показатель, в котором отображена концентрация достижений всех этапов и составляющих учебно-воспитательного процесса. Возможные уровни освоения обучающимся образовательных результатов по программе - низкий (Н), средний (С), высокий (В).

Оценка уровня освоения программы осуществляется по **следующим параметрам и критериям.**

Высокий уровень освоения программы:

- По показателю теоретической подготовки: обучающийся освоил практически весь объём знаний 100-80%, предусмотренных программой за конкретный период; специальные термины употребляет осознанно и в полном соответствии с их содержанием;
- По показателю практической подготовки: обучающийся овладел на 100-80% предметными умениями, навыками и метапредметными учебными действиями, предусмотренными программой за конкретный период; работает с оборудованием самостоятельно, не испытывает особых трудностей; самостоятельно выполняет практические задания с элементами творчества;
- По показателю творческой активности: обучающийся проявляет ярко выраженный интерес к творческой деятельности, к достижению наилучшего результата, коммуникабелен, активен, склонен к самоанализу, генерирует идеи, является участником и призером конкурсных мероприятий городского и выше уровня.

Средний уровень освоения программы:

- По показателю теоретической подготовки: у обучающегося объём усвоенных знаний составляет 79-50%; сочетает специальную терминологию с бытовой;
- По показателю практической подготовки: у обучающегося объём усвоенных предметных умений, навыков и метапредметных учебных действий составляет 79-50%; работает с оборудованием с помощью педагога; в основном, выполняет задания на основе образца;
- По показателю творческой активности: обучающийся имеет устойчивый интерес к творческой деятельности, стремится к выполнению заданий педагога, к достижению результата в обучении, инициативен, является участником конкурсного мероприятия учрежденческого уровня.

Низкий уровень освоения программы:

- По показателю теоретической подготовки: обучающийся овладел менее чем 50% объёма знаний, предусмотренных программой; как правило, избегает употреблять специальные термины;
- По показателю практической подготовки: обучающийся овладел менее чем 50%, предусмотренных предметных умений, навыков и метапредметных учебных действий; испытывает серьёзные затруднения при работе с оборудованием; в состоянии выполнять лишь простейшие практические задания с помощью педагога;
- По показателю творческой активности: обучающийся пассивен, безынициативен, со сниженной мотивацией, нет стремления к совершенствованию в выбранной сфере деятельности, не может работать самостоятельно, отказывается участвовать в конкурсных мероприятиях.

Подведение итогов реализации программы.

В соответствии с календарно-тематическим планом в конце учебного года проводится итоговая аттестация обучающихся в форме презентации итоговых проектов.

Сведения о проведении и результатах итоговой аттестации, обучающихся фиксируются педагогом в журнале учета работы педагога дополнительного образования.

По окончании обучающиеся получают свидетельства об освоении дополнительной образовательной программы «Прикладное программирование». Обучающиеся с высоким и средним уровнем освоения программы получают рекомендацию к обучению на проектном модуле IT-квантума.

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

| № п/п | Наименование раздела | Всего | Из них | |
|-------|--|-----------|----------|----------|
| | | | Теория | Практика |
| | Кейс №1 «Симулятор броска игральных костей» | 8 | 4 | 4 |
| 1 | Введение в курс. Инструктаж по ТБ. Знакомство со средой программирования PyCharm IDE. Основы ввода-вывода. данных. Типы данных | 2 | 1 | 1 |
| 2 | Переменные. Строковые методы. Генерация псевдослучайных чисел. Цикл while | 2 | 1 | 1 |
| 3 | Реализация «Симулятора броска игральных костей». | 2 | 1 | 1 |
| 4 | Презентация результатов выполненного кейса. | 2 | 1 | 1 |
| | Кейс №2 «Крестики-нолики» | 10 | 3 | 7 |
| 5 | Функции. Параметры и возвращаемые значения. | 2 | 1 | 1 |
| 6 | Локальные и глобальные переменные. | 2 | 1 | 1 |
| 7 | Создание плана игры «Крестики-нолики». | 2 | | 2 |
| 8 | Написание функций для игрока и компьютера. Запуск и отладка игры. | 2 | | 2 |
| 9 | Презентация результатов выполненного кейса. | 2 | 1 | 1 |
| | Кейс №3 «Установи соответствие» | 12 | 6 | 6 |
| 10 | Что такое Tkinter. Основы Tkinter и создание окон. | 2 | 1 | 1 |
| 11 | Виджеты в Tkinter. Кнопки, радиокнопки и флажки. Объект Canvas. Позиционирование виджетов. | 2 | 1 | 1 |
| 12 | Разработка программы «Установи соответствие». Создание основного окна и импорт библиотек. | 2 | 1 | 1 |
| 13 | Разработка программы «Установи соответствие». Добавление символов, определение кнопок | 2 | 1 | 1 |
| 14 | Разработка программы «Установи соответствие». Показ символов. | 2 | 1 | 1 |
| 15 | Презентация итогового результата кейса | 2 | 1 | 1 |
| | Кейс №4 «Текстовый редактор» | 10 | 4 | 6 |

| | | | | |
|----|---|-----------|----------|-----------|
| 16 | Текстовый редактор. Создание основного окна. Добавление полосы прокрутки. | 2 | 1 | 1 |
| 17 | Текстовый редактор. Главное меню. Создание пунктов меню «Файл» и «Вид». | 2 | 1 | 1 |
| 18 | Текстовый редактор. Добавление темы оформления окна текстового редактора. | 2 | 1 | 1 |
| 19 | Текстовый редактор. Добавление функций сохранения в файл/загрузки из файла. | 2 | 1 | 1 |
| 20 | Презентация итогового результата кейса. | 2 | | 2 |
| | Кейс №5 «PyGame. Классический space shooter» | 14 | 7 | 7 |
| 21 | PyGame. Основные сведения. Игровой цикл. Шаблон для создания игр. Работа с цветом. Работа со спрайтами. | 2 | 1 | 1 |
| 22 | Спрайт игрока и управление. Спрайты противников. Столкновения с игровыми объектами и выстрелы. | 2 | 1 | 1 |
| 23 | Добавление графики. Улучшение столкновений с игровыми объектами. | 2 | 1 | 1 |
| 24 | Анимация спрайтов. Отображение счёта в игре. Звуковое и музыкальное оформление. | 2 | 1 | 1 |
| 25 | Добавление защитного поля объекту игрока. Добавление анимации взрывов объектов астероидов. | 2 | 1 | 1 |
| 26 | Добавление «полоски жизни» объекта игрока. Добавление выпадающих «бонусов», атаки, защиты объекта игрока. Добавление экрана завершения игры. Отладка и запуск итогового варианта кейса. | 2 | 1 | 1 |
| 27 | Презентация итогового результата кейса | 2 | 1 | 1 |
| | Проектный модуль «Предакселерация проектов» | 18 | 7 | 11 |
| 28 | Поиск идеи. | 2 | 1 | 1 |
| 29 | Целевая аудитория и целевые рынки. Технология работы с гипотезой. | 2 | 1 | 1 |
| 30 | Анализ конкурентов | 2 | 1 | 1 |

| | | | | |
|----|--|-----------|-----------|-----------|
| 31 | Ценностное предложение моего проекта | 2 | 1 | 1 |
| 32 | Способы установления цены модель монетизации | 2 | 1 | 1 |
| 33 | Как подготовиться к экспертной сессии. Структура презентации для экспертов, требования к презентации | 2 | 1 | 1 |
| 34 | Репетиция питчей. | 2 | 1 | 1 |
| 35 | Экспертная сессия – ПИТЧ. | 2 | | 2 |
| 36 | Завершение, рефлексия. | 2 | | 2 |
| | Итого: | 72 | 33 | 39 |

СОДЕРЖАНИЕ

Кейс №1 «Симулятор броска игральных костей» 8 часов / 4 занятия

Цель: знакомство с языком программирования Python и средами программирования на его основе. **Проблемные вопросы:** Какие существуют среды программирования на Python? Как осуществляется ввод-вывод данных? Перечислите типы данных Python? Как инициализировать переменную? Какие существуют методы работы со строками? Какие существуют циклы в Python?

Содержание: в кейсе рассматриваются начальные аспекты работы с языком программирования Python, и средой PyCharm IDE. Особенности создания программ на языке Python, способы их отладки и запуска. Основные типы данных, особенности инициализации переменных в Python. Методы работы со строками. Цикл с неизвестным количеством повторений.

Этапы:

I. Особенности и виды языков программирования, язык программирования Python, особенности разработки программ на Python, виды сред разработки, среда разработки PyCharm IDE. На этом этапе обучающиеся знакомятся с видами и особенностями языков программирования, в частности с особенностями языка программирования Python. Обучающиеся изучают особенности разработки на языке программирования Python.

II. Обучающиеся знакомятся с особенностями ввода-вывода данных на языке программирования Python, с понятием «переменная», понятием «цикл с известным числом повторений», «цикл с неизвестным числом повторений» применительно к языку программирования Python, типами данных и строковыми методами Python.

Гибкие компетенции:

- умение находить, анализировать и использовать информацию
- умение высказывать свою точку зрения

Предметные компетенции:

- иметь представление о типах языков программирования
- иметь представление о языке Python и средах разработки на нём
- иметь основные навыки разработки программ в среде разработки PyCharm IDE
- умение использовать переменные в процессе разработки на Python
- умение использовать строковые методы в процессе разработки на Python

Кейс №2 «Крестики-нолики» 10 часов / 5 занятий

Цель: познакомить обучающихся с глобальными и локальными переменным, функциями и способами их реализации на языке программирования Python, а также с реализацией логики программ с искусственным интеллектом на этом языке.

Проблемные вопросы: что такое функция и как она реализуется на языке Python? Что такое переменные? Какие они бывают? Чем отличаются локальные переменные от глобальных?

Содержание: обучающимся предлагается разработать игру «Крестики-нолики», в ходе которой они познакомятся с понятием «переменная», её типами, с понятием «функция» и особенностями их реализации на языке программирования Python. Дополнительно обучающиеся познакомятся с основами реализации алгоритмов искусственного интеллекта в ходе разработки данной игры.

Этапы:

I. Знакомство обучающихся с понятием «функция», её параметрами, передаваемыми параметрами и возвращаемыми значениями, а также с особенностями их реализации средствами языка программирования Python. Дополнительно обучающиеся знакомятся с понятием «переменная» и их типами: локальная переменная и глобальная переменная, а также особенностями реализации алгоритма искусственного интеллекта на примере игры «Крестики-нолики».

II. Реализация алгоритмов игры «Крестики-нолики» на языке программирования Python.

Гибкие компетенции:

- умение находить, анализировать и использовать информацию;
- умение высказывать свою точку зрения;
- умение работать в группе.

Предметные компетенции:

- умение объявлять переменные на языке программирования Python;
- умение создавать функции, передавать в них параметры, а также обрабатывать возвращаемые значения;
- умение реализовывать алгоритм искусственного интеллекта, отвечающего действиям игрока средствами языка программирования Python.

Кейс №3 «Установи соответствие» 12 часов / 6 занятий

Цель: познакомить обучающихся с библиотекой создания графических интерфейсов пользователя (Graphic User Interface, GUI) на Python на примере создания приложения «Установи соответствие». **Проблемные вопросы:** как создаются окна в Tkinter? Что такое «Виджеты», виды виджетов, взаимное расположение виджетов? Как программно реализовать на Python виджеты кнопок?

Содержание: в рамках кейса обучающиеся создают приложение с графическим интерфейсом пользователя «Установи соответствие» на языке программирования Python. В процессе создания, обучающиеся знакомятся с основными элементами библиотеки Tkinter.

Этапы:

- I. Знакомство с библиотекой Tkinter и её основными элементами, виджетами и их видами.
- II. Программная реализация алгоритма приложения «Установи соответствие» на языке программирования Python.

Гибкие компетенции:

- умение находить, анализировать и использовать информацию;
- умение высказывать свою точку зрения;
- умение работать в группе;

Предметные компетенции:

- умение создавать графические интерфейсы пользователя, используя средства библиотеки Tkinter
- умение составлять и отлаживать программы;

Кейс №4 «Текстовый редактор». 10 часов / 5 занятий

Цель: познакомить обучающихся с процессом создания программы – текстового редактора, похожего на текстовый редактор «Блокнот» ОС Windows или его аналог, используя графические виджеты библиотеки Tkinter.

Проблемные вопросы: как создаётся основное окно текстового редактора? Каким образом отобразить полосу прокрутки в окне текстового редактора? Как добавить в окно текстового редактора пункты основного меню и создать иерархию пунктов подменю? Какой виджет Tkinter отвечает за отображение сообщения о подтверждении выхода из текстового редактора? Каким образом изменяется оформление внешнего вида окна текстового редактора?

Содержание: в рамках кейса обучающиеся создают графическое приложение «Текстовый редактор» на языке программирования Python. В процессе создания приложения,

обучающиеся знакомятся с виджетами библиотеки Tkinter и их использованием в процессе создания приложения.

Этапы:

- I. Создание основного окна текстового редактора, добавление поля для ввода текста, задание абзацных отступов, правил переноса слов, цвета текста и фона, цвета выделенного текста. Добавление полосы прокрутки основного окна текстового редактора;
- II. Добавление главного меню окна текстового редактора. Создание иерархии пунктов меню «Файл», «Вид» окна текстового редактора;
- III. Добавление функций смены темы оформления основного окна текстового редактора и выбора шрифта набираемого текста. Добавление этих функций в команды подменю «Вид» окна текстового редактора;
- IV. Реализация возможности сохранения в файл и загрузки данных из файла.

Гибкие компетенции:

- умение находить, анализировать и использовать информацию;
- умение высказывать свою точку зрения;
- умение работать в группе

Предметные компетенции:

- умение работать с виджетом «Text», отвечающего за ввод текста с клавиатуры, библиотеки Tkinter и задавать его основные параметры: цвет фона, цвет текста, возможность переноса слов по словам, цвет курсора и его внешний вид, цвет выделения набранного текста.
- умение работать с виджетом «Scrollbar» для отображения полосы прокрутки основного окна текстового редактора;
- умение работать с виджетом «Menu», позволяющим создавать основное меню окна текстового редактора, а также создавать иерархию пунктов выпадающего меню;
- умение работать с параметрами оформления окна текстового редактора, заднего фона и шрифта набираемого текста;
- умение использовать такие структуры данных как словарь, функции для реализации реакций на выбор одного из пунктов основного меню окна текстового редактора;
- умение использовать виджет «Messagebox» для отображения сообщения о подтверждении

закрытия основного окна текстового редактора;

- умение использовать виджет «FileDialog» для реализации возможности открытия и сохранения набранного текста в текстовый файл и загрузки текста в окно редактора из файла.

Кейс №5 «PyGame. Классический space shooter» 14 часов / 7 занятий

Цель: познакомить обучающихся с библиотекой PyGame для создания компьютерных игр на языке Python

Проблемные вопросы: как реализуется основной игровой цикл в PyGame? Как в PyGame осуществляется работа с игровыми спрайтами, цветом? Каким образом в игре реализовано столкновение объекта игрока с другими игровыми объектами? Как подсчитываются очки в игре?

Как реализовано выпадение «бонусов», повышающих атакующие и защитные возможности спрайта- игрока?

Содержание: в рамках кейса обучающиеся создают 2D-игру «Классический space shooter», используя возможности и средства языка программирования Python и библиотеки PyGame.

Гибкие компетенции:

- умение находить, анализировать и использовать информацию;
- умение высказывать свою точку зрения;
- умение работать в группе;

Предметные компетенции:

- умение работать с библиотекой PyGame;
- умение создавать и отлаживать программы.

Проектный модуль «Предакселерация проектов» 18 часов / 9 часов

Цель: помочь детским инженерным проектным командам просчитать экономику своих проектов, уточнить гипотезу, а также проанализировать конкурентов, на их основе сформулировать конкурентные преимущества своих проектов и ценностное предложение для возможных инвесторов и партнёров разрабатываемого проекта.

Проблемные вопросы: какая целевая аудитория у разрабатываемого проекта? Какие затраты необходимы для его полноценной работы на первом этапе запуска? Какую рыночную нишу будет занимать разрабатываемый проект? Какие есть конкуренты у разрабатываемого проекта? Кто они?

Содержание.

Обучающимся предлагается проанализировать экономическую сторону разрабатываемых ими проектов с использованием возможностей языка программирования Python. В процессе изучения модуля, обучающиеся уточняют конкретику выбранных ими идей своих проектов, степень полезности таких идей с точки зрения пользы и финансовых результатов, привлекательность выбранной идеи проекта для потенциальных инвесторов и партнёров. Также ребята учатся работать с выдвинутыми гипотезами, проверять их на возможность реализации в виде законченного минимального жизнеспособного продукта (MVP), производить анализ целевых рынков, формировать ценностное предложение. Кроме того, ребята узнают о способах монетизации своих проектов и влияние их на цену и себестоимость. Для обучающихся 13-17 лет можно предложить примерные темы проектов для самостоятельной разработки.

Примерный перечень проектов:

1. Программа для тестирования знаний по выбранному предмету.
2. Приложение-тренажёр для отработки каких-либо навыков.
3. Игровое приложения в виде текстового квеста или игры.
4. Приложение для воспроизведения музыки(MP3-проигрыватель)

В процессе работы, обучающиеся знакомятся с основами Scrum-метода организации эффективной работы над проектом.

Этапы:

- I. Обучающиеся организуются в команды, проводят мозговой штурм, выбирают тему проекта. По выбранной ими идее проводят поиск информации, анализируют целевую аудиторию, работают с выдвинутыми в процессе разработки проекта гипотезами на их применимость и возможность реализации в виде минимального жизнеспособного продукта (MVP), осуществляют целеполагание и планирование.
- II. Проектируют сюжет/сценарий исполнения проекта, распределяют работу по проекту
- III. Обучающиеся анализируют конкурентов, выявляют их сильные и слабые стороны. На основе произведённого анализа формируют основные конкурентные преимущества своих проектов.
- IV. Обучающиеся создают программный код, проводят тестирование и отладку приложения.
- V. Обучающиеся готовят презентацию проекта и демонстрируют все возможности приложения.
- VI. Обучающиеся изучают способы установления цены и модель монетизации своих проектов, готовят презентацию проекта в соответствии с требованиями для презентации

для экспертов, потенциальных инвесторов и партнёров. Демонстрируют все возможности разработанного проекта. Готовятся к экспертной сессии-ПИТЧу.

- VII. Обучающиеся готовят речь для своего выступления на экспертной сессии-ПИТЧе перед инвесторами и партнёрами.
- VIII. Обучающиеся выступают на экспертной сессии.
- IX. С обучающимися проводится рефлексия по итогам их выступлений на экспертной сессии-ПИТЧе.

РЕСУРСНОЕ ОБЕСПЕЧЕНИЕ

Календарный график

Количество учебных недель по программе – 36.

Количество учебных дней по программе – 36.

Каникул нет.

Начало учебного года – 1 сентября, окончание – 31 мая.

Календарно-тематический план представлен в Приложении 1.

Методическое обеспечение

1. Педагогические технологии, методы, приемы и формы организации образовательного процесса

При реализации программы используются следующие педагогические технологии:

- технологии развивающего обучения, направленные на общее целостное развитие личности, на основе активно-деятельного способа обучения, учитывающие закономерности развития и особенности индивидуума;
- технологии личностно-ориентированного обучения, направленные на развитие индивидуальных познавательных способностей каждого ребёнка, максимальное выявление, раскрытие и использование его опыта;
- технологии дифференцированного обучения, обеспечивающие обучение каждого обучающегося на уровне его возможностей и способностей;
- технологии сотрудничества, реализующие демократизм, равенство, партнёрство в отношениях педагога и обучающегося, совместная выработка цели, содержания, оценивание, находясь в состоянии сотрудничества, сотворчества;
- проектные технологии – достижение цели через детальную разработку проблемы, которая должна завершиться реальным, осязаемым практическим результатом, оформленным тем или иным образом;
- компьютерные технологии, формирующие умение работать с информацией, исследовательские умения, коммуникативные способности.

В практике выступают различные комбинации этих технологий, их элементов.

Основным методом организации учебной деятельности по программе является метод кейсов.

Кейс – описание проблемной ситуации понятной и близкой обучающимся, решение

которой требует всестороннего изучения, поиска дополнительной информации и моделирования ситуации или объекта, с выбором наиболее подходящего.

Преимущества метода кейсов

- Практическая направленность. Кейс-метод позволяет применить теоретические знания к решению практических задач.
- Интерактивный формат. Кейс-метод обеспечивает более эффективное усвоение материала за счет высокой эмоциональной вовлеченности и активного участия обучаемых.
- Конкретные навыки. Кейс-метод позволяет совершенствовать «гибкие навыки» (soft skills), которым не учат в университете, но которые оказываются крайне необходимы в реальном рабочем процессе.
- В ходе работы над кейсом целесообразно использовать следующие методы, приемы, средства и формы организации, внесённые в таблицу.

| № | Формы организации | Методы и приемы | Дидактический материал | Формы контроля |
|----------|--|---|---|--|
| 1 | Лекция с разбором решения практического задания | устное изложение с визуальным рядом, позволяющий в доступной форме донести до обучающихся материал; | Видео, приложения, шаблоны файлов, веб-доски и веб-плакаты. | Проверка синхронного выполнения материала лекции. |
| 2 | Обсуждение, рефлексия рассмотренных тем | Устный опрос в ходе демонстрации видеоряда | Видео-презентация | рефлексивный самоанализ, контроль и самооценка обучающихся |
| 3 | Практическое задание, сходное с разбиравшимся на лекции; | репродуктивный практический метод; частично-поисковый | Видео, шаблоны исходных файлов, веб-плакаты | Просмотр хода выполнения; обсуждение итогов |
| 4 | Проект | исследовательский метод практический метод частично-поисковый | Веб-доски и веб-документы, видео, инструкции по работе над проектом, шаблоны файлов | Защита проекта |
| 5 | Соревнование | практический метод | Веб-доски и веб-документы | Подведение итогов. |
| 6 | Исследование | исследовательский метод | Презентация, видео, описание хода исследования и т.д. | Конференция |

2. Учебно-методический комплекс программы

Для реализации программы используются:

- авторские учебные видео и презентации, раздаточный материал к каждой теме;
- специализированная литература по программированию, подборка журналов;
- наборы технической документации к применяемому оборудованию;
- образцы игровых приложений, выполненных обучающимися и педагогом;
- плакаты, фото и видеоматериалы,
- учебно-методические пособия для педагога и обучающихся, включающие дидактический, информационный, справочный материалы на различных носителях, компьютерное и видео оборудование.

Методические материалы для педагога

1. Методические рекомендации, конспекты занятий, сценарии мероприятий, памятки:
 - 1.1. Тулжит IT-квантума
 - 1.2. Памятки по темам программы.
 - 1.3. Практические работы по темам программы.
2. Диагностический инструментарий:
 - 2.1. Входная диагностика первого обучения
 - 2.2. Тесты для текущего контроля знаний.
 - 2.4. Анкета для родителей «Удовлетворенность результатами посещения ребенком занятий объединения».
3. Организационно-методические материалы:
 - 3.1. Календарно-тематическое планирование учебного материала на учебный год;
 - 3.2. Инструкции по охране труда и технике безопасности.
 - 3.3. Положения, приказы, информационные письма о проведении мероприятий различного уровня по профилю объединения.

3. Информационное обеспечение

Литература для обучающихся

1. Доусон М. Програмируем на Python. – СПб.: Питер, 2014 г. – 416 с
2. Келли С. Разработка игр на PyGame, Raspberry Pi – 2016г. – 200 с
3. Эрик Мэтиз. Изучаем Python. Программирование игр, визуализация данных, веб-приложения. – СПб: Питер, 2020 – 512 с

Используемые интернет-ресурсы

| № | Интернет-адрес | Название ресурса | Где используется и для чего |
|---|---|--|-----------------------------|
| 1 | https://docs.python.org/3/tutorial/index.html | Официальный обучающий ресурс по языку программирования Python | Разделы 1-6 |
| 2 | https://kidscancode.org/lessons/ | Обучающий ресурс по разработке игровых приложений средствами библиотеки PyGame | Разделы 1-6 |
| 3 | https://www.youtube.com/channel/UC_aTa7Q7gFN5eaKKmGO8jXg | YouTube-канал «Программирование Создание игр, сайтов и т.д.» | |
| | | | |

СПИСОК ЛИТЕРАТУРЫ,

использованной при составлении данной программы

Нормативные документы

1. Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации» (с изменениями от 29.12.2022 N 642-ФЗ)
2. Концепция развития дополнительного образования детей до 2030 года (утверждена распоряжением Правительства РФ от 31.03.2022 № 678-р)
3. Приказ Министерства просвещения РФ от 27.07.2022 № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»
4. Постановление Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»
5. Указ Президента Российской Федерации от 21 июля 2020 года № 474 «О национальных целях развития Российской Федерации на период до 2030 года»
6. Стратегия развития воспитания в Российской Федерации на период до 2025 года (утверждена распоряжением Правительства Российской Федерации от 29.05.2015 № 996-р)
7. Методические рекомендации по разработке дополнительных общеобразовательных программ. Письмо Министерства образования и науки Самарской области от 03.09.2015 г. № МО-16-09-01/826-ту [Электронный ресурс] / Самарский дворец детского и юношеского творчества. - Режим доступа: <http://pioner-samara.ru/content/metodicheskaya-deyatelnost> .
8. Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы). Письмо Департамента государственной политики в сфере воспитания детей и молодежи Министерства образования и науки РФ № 09-3242 от 18.11.2015 г. [Электронный ресурс] / Самарский дворец детского и юношеского творчества. – Режим доступа: pioner-samara.ru/sites/default/files/docs/metodrek_dop_rf15.doc.

Общепедагогическая, психологическая и методическая литература

1. Буйлова, Л.Н. Методические рекомендации по подготовке авторских программ дополнительного образования детей / Л.Н. Буйлова, Н.В. Кленова, А.С. Постников [Электронный ресурс] / Дворец творчества детей и молодежи. В помощь педагогу. – Режим доступа: <http://doto.ucoz.ru/metod/>.

2. Гин, А.А. Приёмы педагогической техники: свобода выбора, открытость, деятельность, обратная связь, идеальность: Пособие для учителей / А.А. Гин. – Гомель : ИПП «Сож», 1999. – 88 с.
3. Конасова, Н.Ю. Оценка результатов дополнительного образования детей. ФГОС. / Н.Ю. Конасова. - Волгоград: Учитель, 2016. – 121с. – (Образовательный мониторинг).

Специальная литература

1. Доусон М. Програмируем на Python. – СПб.: Питер, 2014 г. – 416 с
2. Келли С. Разработка игр на PyGame, Raspberry Pi – 2016г. – 200 с
3. Эрик Мэтиз. Изучаем Python. Программирование игр, визуализация данных, веб-приложения. – СПб: Питер, 2020 – 512 с
4. Кучма, В.Р. Гигиена детей и подростков при работе с компьютерными видеодисплейными терминалами. / В.Р. Кучма. - М. : Медицина, 2000. - 160 с.

ПРИЛОЖЕНИЕ 1. КАЛЕНДАРНО-ТЕМАТИЧЕСКИЙ ПЛАН

| Срок | № п/п | Наименование раздела | Часов всего | Количество часов | | Форма контроля |
|--------------------------|-------|---|----------------|---------------------|----------|-------------------------------|
| | | | | Теория | Практика | |
| | | Кейс №1 «Симулятор броска игральных костей» | 8 | 4 | 4 | |
| 04 – 10 сентября | 1 | Введение в курс. Инструктаж по ТБ. Знакомство со средой программирования PyCharm IDE. Основы ввода-вывода данных. Типы данных | 2 | 1 | 1 | тестирование |
| 11 – 17 сентября | 2 | Переменные. Строковые методы. Генерация псевдослучайных чисел. Цикл while | 2 | 1 | 1 | тестирование |
| 18 – 24 сентября | 3 | Реализация «Симулятора броска игральных костей» | 2 | 1 | 1 | демонстрация работы программы |
| 25 сентября – 01 октября | 4 | Презентация результатов выполненного кейса | 2 | 1 | 1 | презентация |
| | | Кейс №2 «Крестики-нолики» | 10 | 3 | 7 | |
| 02 – 08 октября | 5 | Функции. Параметры и возвращаемые значения. | 2 | 1 | 1 | тестирование |
| 09 – 15 октября | 6 | Локальные и глобальные переменные. | 2 | 1 | 1 | тестирование |
| 16 – 22 октября | 7 | Создание плана игры «Крестики-нолики» | 2 | | 2 | контрольные вопросы |
| 23 – 29 октября | 8 | Написание функций для игрока и компьютера. Запуск и отладка игры | 2 | | 2 | демонстрация работы программы |
| 30 октября – 05 ноября | 9 | Презентация результатов выполненного кейса | 2 | 1 | 1 | презентация |
| | | Кейс №3 «Установи соответствие» | 12 | 6 | 6 | |
| 06 - 12 ноября | 10 | Что такое Tkinter. Основы Tkinter и создание окон. | 2 | 1 | 1 | тестирование |
| 13 - 19 ноября | 11 | Виджеты в Tkinter. Кнопки, радиокнопки и флажки. Объект Canvas. Позиционирование виджетов. Методы Pack, Grid, Place. | 2 | 1 | 1 | тестирование |
| 20 - 26 ноября | 12 | Разработка программы «Установи соответствие». Создание основного окна и импорт библиотек | 2 | 1 | 1 | демонстрация работы программы |

| | | | | | | |
|------------------------|----|--|-----------|----------|----------|-------------------------------|
| 27 ноября – 03 декабря | 13 | Разработка программы «Установи соответствие». Добавление символов, определение кнопок | 2 | 1 | 1 | демонстрация работы программы |
| 04 – 10 декабря | 14 | Разработка программы «Установи соответствие». Показ символов. Завершение | 2 | 1 | 1 | демонстрация работы программы |
| 11 – 17 декабря | 15 | Презентация итогового результата кейса | 2 | 1 | 1 | презентация |
| | | Кейс №4 «Текстовый редактор» | 10 | 4 | 6 | |
| 18 – 24 декабря | 16 | Текстовый редактор. Создание основного окна. Добавление полосы прокрутки | 2 | 1 | 1 | демонстрация работы программы |
| 25 – 31 декабря | 17 | Текстовый редактор. Главное меню. Создание пунктов меню «Файл» и «Вид» | 2 | 1 | 1 | демонстрация работы программы |
| II полугодие | | | | | | |
| 8 - 14 января | 18 | Текстовый редактор. Добавление темы оформления окна текстового редактора | 2 | 1 | 1 | демонстрация работы программы |
| 15 - 21 января | 19 | Текстовый редактор. Добавление функций сохранения в файл/загрузки из файла. | 2 | 1 | 1 | демонстрация работы программы |
| 22 - 28 января | 20 | Презентация итогового результата кейса | 2 | | 2 | Презентация |
| | | Кейс №5 «PyGame. Классический space shooter» | 14 | 7 | 7 | |
| 29 января- 04 февраля | 21 | PyGame. Основные сведения. Игровой цикл. Шаблон для создания игр. Работа с цветом. Работа со спрайтами | 2 | 1 | 1 | демонстрация работы программы |
| 05-11 февраля | 22 | Спрайт игрока и управление. Спрайты противников. Столкновения с игровыми объектами и выстрелы. | 2 | 1 | 1 | демонстрация работы программы |
| 12 - 18 февраля | 23 | Добавление графики. Улучшение столкновений с игровыми объектами | 2 | 1 | 1 | демонстрация работы программы |
| 19 - 25 февраля | 24 | Анимация спрайтов. Отображение счёта в игре. Звуковое и музыкальное оформление | 2 | 1 | 1 | демонстрация работы программы |
| 26 февраля - 03 марта | 25 | Добавление защитного поля объекту игрока. Добавление анимации взрывов объектов | 2 | 1 | 1 | демонстрация работы программы |

| | | | | | | |
|-----------------------|----|--|-----------|-----------|-----------|--|
| 04 - 10 марта | 26 | Дополнительные возможности оформления. | 2 | 1 | 1 | демонстрация работы программы |
| 11- 17 марта | 27 | Презентация итогового результата кейса. | 2 | 1 | 1 | Презентация |
| | | Проектный модуль «Предакселерация проектов» | 18 | 7 | 11 | |
| 18 - 24 марта | 28 | Поиск идеи. | 2 | 1 | 1 | демонстрация решения готового этапа |
| 25-31 марта | 29 | Целевая аудитория и целевые рынки. Технология работы с гипотезой. | 2 | 1 | 1 | демонстрация решения готового этапа |
| 01-07 апреля | 30 | Анализ конкурентов | 2 | 1 | 1 | демонстрация решения готового этапа |
| 08 - 14 апреля | 31 | Ценностное предложение моего проекта | 2 | 1 | 1 | демонстрация решения готового этапа |
| 15 - 21 апреля | 32 | Способы установления цены модель монетизации | 2 | 1 | 1 | демонстрация решения готового этапа |
| 22-28 апреля | 33 | Как подготовиться к экспертной сессии. Структура презентации для экспертов, требования к презентации | 2 | 1 | 1 | демонстрация решения готового этапа |
| 29 апреля – 05 мая | 34 | Репетиция питчей | 2 | 1 | 1 | демонстрация решения готового этапа |
| 06 – 12 мая | 35 | Экспертная сессия – ПИТЧ | 2 | | 2 | выступление команд учащихся перед экспертами |
| 13-19 мая | 36 | Завершение, рефлексия | 2 | | 2 | рефлексия с командами учащихся |
| | | Итого: | 72 | 31 | 41 | |

ПРИЛОЖЕНИЕ 2. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

Входная диагностика

Входная диагностика проводится на первом занятии.

Ход проведения диагностики

Обучающиеся разбиваются на группы (пары) и выполняют задания. Использовать можно любые источники: Интернет, собственные знания, учебные пособия, которые находятся в кабинете. Время выполнения заданий ограничено. По окончании обучающиеся презентуют результаты своей работы. Если ребенок не хочет работать в группе, можно разрешить ему выполнение заданий индивидуально, отразив это в диагностической карте в метрике «Умение работать в команде».

Вопросы:

1. Опишите основные принципы объектно-ориентированного программирования так, чтобы их понял учащийся 4-5 класса.

2. Классифицируйте самые популярные языки программирования. Составьте два списка не менее чем по 5 позиций в каждом. В первом списке напишите объектно-ориентированные языки, во втором, языки программирования, которые к объектно-ориентированным не относятся.

3. Ответ на вопросы 1-2 оформите в презентацию.

Наставник методом наблюдения определяет уровень предметных и гибких навыков, определяя их как высокий, средний и низкий. Результат диагностики заносится в карту.

Примерный вид диагностической карты

| | |
|--|-----------|
| ФИО | |
| Метрика | Уровень |
| Умение осуществлять эффективный поиск информации | В / С / Н |
| Общая предметная осведомленность | В / С / Н |
| Умение работать в команде | В / С / Н |
| Умение презентовать выполнение задания | В / С / Н |

Итоговая аттестация

При подготовке к защите проекта учащимся необходимо подготовить презентацию и доклад, в котором отражаются основные этапы работы над проектом, основные результаты работы. Можно предложить в помощь обучающимся заполнить следующий шаблон:

1. Аннотация.
2. Содержание.
3. Постановка задачи:
 - а. актуальность и проблематика проекта
 - б. исследование существующих аналогов
4. Описание проекта:
 - а. техническое задание
 - б. описание необходимых ресурсов
 - с. планирование работы по проекту
5. Тестовые примеры
 - а. результаты работы по проекту
 - б. скриншоты/фото результатов работы
 - с. пути улучшения результатов

Лист оценивания проекта

| <i>Критерий оценивания</i> | <i>Группа 1</i> | <i>Группа 2</i> | <i>...</i> |
|---|-----------------|-----------------|------------|
| Актуальность проекта | | | |
| Соответствие содержания проекта заявленной проблематике | | | |
| Техническая сложность разработанного устройства/решения | | | |
| Оригинальность устройства/решения | | | |
| Степень разработанности устройства/решения | | | |
| Итоговое количество баллов | | | |