

**Министерство образования, науки и молодежной политики
Нижегородской области
Государственное бюджетное учреждение дополнительного образования
«Центр молодежных инженерных и научных компетенций
«КВАНТОРИУМ»**

РАССМОТРЕНА
на Совете коллектива
ГБУДО «ЦМИНК «Кванториум»
Протокол от «12» 03 2019 г.
№ 3

УТВЕРЖДАЮ
и.о. директор
ГБУДО «ЦМИНК «Кванториум»
С.А.Непокорова
«12» 03 2019 г.



IT-КВАНТУМ

**Дополнительная общеобразовательная общеразвивающая программа
«IT -Квантум»**

(Вводный модуль - линия 0; Базовый модуль – линия 1)

Направленность: научно-техническая

Возраст обучающихся: 13 - 17 лет

Длительность Вводного модуля: 72 часа

Длительность Базового модуля: 72 часа

Всего: 144 часа

Автор: Маковкин Сергей Юрьевич,
педагог дополнительного образования

ИНФОРМАЦИОННАЯ КАРТА ПРОГРАММЫ

1	Полное название программы	Дополнительная общеобразовательная общеразвивающая программа «ИТ -Квантум»
2	Автор программы	Маковкин Сергей Юрьевич, педагог дополнительного образования
3	Название образовательной организации	ГБУДО «ЦМИНК «КВАНТОРИУМ»
4	Адрес организации	г. Н. Новгород, ул. Ульянова, 10 Б, к.т. 7 (831) 214-03-55
5	Форма проведения	Групповые и индивидуальные занятия
6	Вид программы по уровню усвоения содержания программы	Развивающая, практико-деятельностная, личностно-ориентированная, проектная. Линия 0 – Вводный модуль Линия 1 – Базовый модуль
7	Цель программы	Формирование ИТ - компетентности школьников; приобретение опыта использования информационных технологий в индивидуальной и коллективной учебной и познавательной, в том числе проектной деятельности.
8	Специализация программы	Информационные технологии
9	Направленность программы	Научно-техническая
10	Сроки реализации	Вводный модуль - 72 часа Базовый модуль – 72 часа
11	Количество участников программы	Группа №1 (13-14 лет) Группа №2 (15-17 лет)
12	География участников программы	г. Нижний Новгород
	Условия участия в программе	Обучающиеся 13-14 и 15-17 лет
13	Условия размещения участников программы	Оборудованная лаборатория детского технопарка «Кванториум»
14	Ожидаемый результат	<u>Обучающиеся 13-14 лет</u> В процессе освоения программы, обучающиеся приобретут знания об операционных системах Unix/Linux, скриптовом Shell-языке bash, о распространённых языках программирования и опыт в использовании языка Python, о видах компьютерных игр, основах визуального языка

	программирования, принципах ввода и вывода данных, принципах управления внутри игровыми объектами; овладеют навыками анализа и проектирования игровых приложений, разработки игровых приложений с использованием конструктора компьютерных игр. В ходе занятий обучающиеся будут вовлечены в проектную деятельность, которая позволит им в малых группах разрабатывать и представлять проекты, они научатся обосновывать свою точку зрения и решать исследовательские задачи. <u>Обучающиеся 15-17 лет</u> В процессе освоения программы, обучающиеся приобретут знания об операционных системах Unix/Linux, скриптовом Shell-языке bash, о распространённых языках программирования и опыт в использовании языка Python и экосистемы библиотек и фреймворков; овладеют навыками анализа и проектирования игровых приложений, разработки игровых и неигровых приложений. В ходе занятий обучающиеся будут вовлечены в проектную деятельность, которая позволит им в малых группах разрабатывать и представлять проекты, научатся обосновывать свою точку зрения и решать исследовательские задачи.
--	---

Содержание

1. Пояснительная записка.....	5
2. Учебно-тематический план IT-START 13-14 лет... ..	12
3. Учебно-тематический план IT-TOTAL 15-17 лет.....	13
4. Содержание программы IT-START 13-14 лет.....	15
5. Содержание программы IT-TOTAL 15-17 лет.....	19
6. Базовый модуль (Линия 1).....	25
7. Учебно-тематический план. Базовый модуль (Линия 1).....	26
8. Содержание программы. Базовый модуль (Линия 1).....	26
9. Учебно-методическое обеспечение программы.....	41
10. Материально-техническое обеспечение.....	43
11. Список рекомендуемой литературы.....	44
12. Приложения.....	46

Пояснительная записка

На сегодняшний день одной из самых востребованных областей на рынке труда является сфера информационных технологий. Жизнь и деятельность в информационном обществе невозможна без знаний и умений в данной области. Эти сотрудники отвечают за настройку сетей, серверов, контролируют их эксплуатацию, разрабатывают программные продукты, алгоритмы и искусственный интеллект.

Стремительный рост информационных технологий ставит новые задачи перед образованием и наукой. В связи с этим актуальной задачей является подготовка специалистов сферы ИТ в соответствии с профессиональными требованиями динамично развивающихся отраслей. При этом требуется постоянная актуализация знаний, приобретение новых компетенций, формирование нового типа мышления.

В этом смысле важнейшую роль играет процесс изучения базовых основ информационных технологий еще в школьном возрасте.

Программа курса направлена на формирование навыков использования информационных технологий в проектной деятельности, что поможет в дальнейшем обучении.

Программа «ИТ - Квантум» разработана педагогом дополнительного образования Маковкиным Сергеем Юрьевичем для реализации на базе детского технопарка «Кванториум» Нижний Новгород. Данная программа создана в рамках стратегической инициативы «Новая модель системы дополнительного образования детей».

Основное назначение программы состоит в выполнении социального заказа современного общества, направленного на подготовку подрастающего поколения к полноценной работе в условиях глобальной информатизации всех сторон общественной жизни.

Направленность программы: научно-техническая.

Актуальность программы дополнительного образования обусловлена тем, что в настоящее время важной задачей современного образования является воспитание нового поколения, которое должно отвечать уровню развития информационного общества. Это достигается через освоение способов работы с информационными потоками, а именно поиск необходимой информации, ее анализ, преобразование в структурированную текстовую форму (программу), использование для решения практических задач. Умение передать информацию в удобной форме для восприятия и использования другими людьми, является условием образовательной компетенции учащегося.

Новизна программы:

Новизна программы заключается в том, что обучение имеет ярко выраженный практический характер, в основе методики обучения лежат кейсовый и проектный методы, технологии изобретательской разминки и идеального конечного результата, научный эксперимент.

Содержание и материал образовательной программы организован по принципу дифференциации в соответствии со следующими уровнями сложности:

«Вводный модуль». Предполагает использование и реализацию общедоступных и универсальных форм организации материала, минимальную сложность предлагаемого для освоения содержания программы, развитие мотивации к определенному виду деятельности.

В Вводном модуле (Линия 0) программы ребята освоят современные языки программирования, популярные и актуальные фреймворки с необходимым функционалом, познакомятся с алгоритмами создания компьютерных игр (hard skills) приобретут навыки, которые очень важны как для участия в коллективных проектах, так и в жизни в социуме: работать совместно, брать на себя нужную для команды роль, нести ответственность, помогать друг другу и т. д. (soft skills).

По окончании Вводного модуля – отчисление. Далее возможно два

пути: зачисление на Базовый модуль (Линия 1) или обучение в Вводном модуле другого направления (квантума).

Программа «Базового модуля» (Линия 1) предусматривает командную работу с использованием знаний и умений, полученных в процессе обучения на этапе Вводного модуля.

Выбор траектории обучения в Базовом модуле предполагает построение индивидуального образовательного маршрута в составе команды от 3 до 7 человек, составленного совместно с обучающимися на основе их предпочтений, и предполагает определенные результаты в виде конкретного ИТ продукта, междисциплинарных проектов, творческих работ, участия в олимпиадах и конкурсах.

Ключевой принцип проектного обучения заключается в ориентации на практическое решение проблем. При этом проблема, на решение которой направлен проект, должна быть подлинной, касающейся реального мира. Так же важная особенность проектного обучения данного модуля – междисциплинарность, требующая формирования разнопрофильных проектных команд.

Все разработанные командами проекты представляются на внутреннем и региональном уровнях, а авторы лучших работ направляются на всероссийские и международные конкурсы и олимпиады.

Педагогическая целесообразность заключается в предоставлении школьнику спектра возможностей по реализации его интересов и способностей в сфере ИТ-технологий, создания самостоятельных творческих работ средствами ИТ-технологий, формировании информационной культуры, обеспечении интегрированного подхода в изучении традиционных учебных предметов, формировании мотивации детей и подростков к изучению и использованию ИТ-технологий с последующим выбором профессии.

Цель программы: сформировать ИТ-компетентность учащихся; приобрести опыт использования информационных технологий в познавательной, учебной проектной деятельности.

Задачи

Обучающие:

- ✓ Сформировать представление о сущности информации и информационных процессов;
- ✓ Познакомить с основными понятиями информатики непосредственно в процессе создания информационного продукта;
- ✓ Сформировать навыки владения операционными системами Unix/Linux, писать Shell-скрипты на языке bash;
- ✓ Сформировать навыки создания компьютерных игр, навыки разработки интерактивных приложений на языке Python;
- ✓ Сформировать навыки использования компьютерной техники как практический инструмент для работы с информацией в учебной деятельности и повседневной жизни;
- ✓ Изучать основы алгоритмизации, построения алгоритмов и их формализации с помощью языка блок-схем;

Развивающие:

- ✓ Развивать познавательные способности обучающегося, память, внимание, пространственное мышление, эстетическое мировоззрение;
- ✓ Развивать логическое и алгоритмическое мышление.
- ✓ Сформировать у обучающихся навыки творческого подхода к поставленной задаче, командной работе и публичных выступлений по IT-тематике;

Воспитательные:

- ✓ Воспитывать усидчивость, умение преодолевать трудности;
- ✓ Сформировать информационную культуру;
- ✓ Сформировать коммуникативные умения;
- ✓ Развивать мотивацию личности к познанию;
- ✓ Сформировать нравственные качества личности и культуру поведения в обществе.

Основные характеристики образовательной программы:

Уровень программы (модуля): Линия 0 - вводный модуль

Линия 1 – базовый уровень

Возраст обучающихся, участвующих в реализации программы:

IT-START (начинающий пользователь ПК, базовые навыки программирования) – обучающиеся **13-14 лет**.

IT-TOTAL (пользователь операционных систем Unix/Linux, продвинутые навыки программирования) - обучающиеся **15-17 лет**.

Срок реализации программы (модуля): Вводный модуль - 72 часа;
Базовый модуль - 72 часа.

Режим занятий: 2 раза в неделю по 2 академических часа.

Формы организации учебной деятельности: групповая, индивидуальная, парная.

Количество обучающихся в группе: 8 - 14 человек.

Прогнозируемые результаты:

Л и ч н о с т н ы е р е з у л ь т а т ы

- ✓ критическое отношение к информации и избирательность ее восприятия;
- ✓ осмысление мотивов своих действий при выполнении заданий;
- ✓ развитие любознательности, сообразительности при выполнении разнообразных заданий проблемного и эвристического характера;
- ✓ развитие внимательности, настойчивости, целеустремленности, умения преодолевать трудности – качеств, весьма важных в проектной деятельности;
- ✓ развитие самостоятельности суждений, независимости и нестандартности мышления;
- ✓ воспитание чувства справедливости, ответственности;
- ✓ начало профессионального самоопределения, ознакомление с миром профессий, связанных с IT-технологиями.

Предметные результаты

Обучающиеся 13-14 лет

В процессе освоения программы, обучающиеся приобретут знания об использовании и работе операционных систем Unix/Linux, распространенных языках программирования и опыт в использовании языка Python, о видах компьютерных игр, принципах ввода и вывода данных, принципах управления внутри игровыми объектами; овладеют навыками анализа и проектирования игровых приложений, разработки игровых приложений с использованием конструктора компьютерных игр. В ходе занятий обучающиеся будут вовлечены в проектную деятельность, которая позволит им в малых группах разрабатывать и представлять проекты, они научатся обосновывать свою точку зрения и решать исследовательские задачи.

Обучающиеся 15-17 лет

В процессе освоения программы, обучающиеся приобретут знания об использовании и работе операционных систем Unix/Linux, распространенных языках программирования и опыт в использовании языка Python и экосистемы библиотек и фреймворков, овладеют навыками анализа и проектирования игровых приложений, разработки игровых и неигровых приложений на языке Python, обработки выходной информации и анализа статистики. В ходе занятий обучающиеся будут вовлечены в проектную деятельность, которая позволит им в малых группах разрабатывать и представлять проекты, они научатся обосновывать свою точку зрения и решать исследовательские задачи.

После прохождения программы, обучающиеся получают:

- ✓ навыки работы в операционных системах Unix/Linux;
- ✓ навыки создания сценариев и Shell-скриптов на языке bash;
- ✓ навыки программирования на языке Python;
- ✓ навыки создания и обработки графических изображений (ImageMagic);
- ✓ навыки работы в издательской системе TeX/LaTeX.

Обучающиеся научатся создавать:

- ✓ интерактивные приложения, обеспечивающие реализацию современных алгоритмов;
- ✓ компьютерные игры с искусственным интеллектом, как пример комплексного цифрового продукта;

Формы диагностики образовательных результатов:

а) входной контроль (педагогическое наблюдение, опрос, анкеты, викторины);

В результате определяются знания по технике безопасности, интересы ребенка, его ожидания.

б) промежуточная аттестация (опрос на основе полученных знаний на текущий момент времени, выполнение кейс-заданий);

Проводится проверка знаний, умений и навыков при помощи разработанных кейсов

в) итоговая аттестация (опрос на основе полученных знаний по итогам модуля, защита проектов)

Формы демонстрации результатов обучения: мини-конференция по защите проектов, выставка, внутригрупповой конкурс (соревнования), презентация (самопрезентация) проектов обучающихся и др., конкурс проектов обучающихся с целью отбора в проектные команды на постоянной основе.

Учебно-тематический план

а. Вводный модуль - линия 0 (IT-START 13-14 лет)

№	Наименование разделов и дисциплин	Всего	Теория	Практика	
				Квантум	Хайтек
1	Вводное занятие. Введение в предмет, техника безопасности	2	1	1	
2	Знакомство с операционной системой Linux Debian 9.6.	4	2	2	
3	Освоение скриптового языка bash (ввод-вывод, архивирование, поиск файлов).	4	2	2	
4	Освоение текстового редактора Vim.	4	2	2	
5	Изучение издательской системы TeX/LaTeX.	4	2	2	
6	Написание и тестирование скриптов для администрирования системы Linux.	4	2	2	
7	Основы языка Python. Примеры на языке Python с разбором конструкций: циклы, условия, ветвления, массивы, типы данных.	4	2	2	
8	Кейс «Числовые загадки»	10	4	6	
8.1.	Оформление проектной идеи. Формирование программы работ.	2	1	1	
8.2.	Введение в искусственный интеллект. Примеры на языке Python с искусственным интеллектом по отгадыванию чисел, метод дихотомии. Управление искусственным интеллектом.	6	2	4	
8.3.	Подготовка к публичному выступлению для защиты результатов. Демонстрация отчета в группе и защита результатов работы.	2	1	1	
9	Кейс «Игра Виселица»	12	5	7	
9.1.	Работа на языке Python со словарями и списками, множественное присваивание, добавление и удаление элементов в список.	4	2	2	
9.2.	Планирование дизайна и механики игры. Создание главного меню игры, подсчета очков.	2	1	1	
9.3.	Визуализирование программы в виде блок-схемы.	2	1	1	
9.4.	Тестирование написанной программы и доработка.	2		2	
9.5.	Подготовка к публичному выступлению для защиты результатов. Демонстрация результатов работы	2	1	1	
10.	Продвинутые конструкции языка Python.	4	2	2	

	Генерация на языке Python двумерного игрового поля, методики перетасовка элементов в массивах, игровой цикл				
11.	Примеры на языке Python на темы: различия между словами и списками, обработка ошибочных предположений, игровое меню, завершение и перезагрузка игры.	6	2	4	
12	Кейс «Игра Крестики-Нолики»	14	1	9	4
12.2.	Текстовая версия с искусственным интеллектом, игра: человек с компьютером, компьютер с компьютером.	2	1	1	
12.3.	Обработка статистики, вывод и обработка результатов.	2		2	
12.4.	Тестирование написанной программы и доработка.	2		2	
12.5.	Оформление презентационной игры. 3D-печать.	4			4
12.6.	Подготовка к публичному выступлению для защиты результатов.	2		2	
12.7.	Демонстрация результатов работы	2		2	
	ВСЕГО:	72	27	41	4

б. Вводный модуль - линия 0 (IT-TOTAL 15-17 лет)

№	Наименование разделов и дисциплин	Всего	Теория	Практика	
				Квантум	Хайтек
1	Вводное занятие. Введение в предмет, техника безопасности	2	1	1	
2	Оформление проектной идеи. Формирование программы работ. Знакомство с операционной системой Linux Debian 9.6.	4	2	2	
3	Освоение скриптового языка bash (ввод-вывод, архивирование, поиск файлов)	4	2	2	
4	Освоение текстового редактора Vim	4	2	2	
5	Изучение издательской системы TeX/LaTeX	4	2	2	
6	Написание и тестирование скриптов для администрирования системы Linux	4	2	2	
7	Основы языка Python. Примеры на языке Python разбором конструкций: циклы, условия, ветвления, массивы, типы данных.	4	2	2	
8	Кейс «Числовые загадки»	10	3	7	
8.2.	Оформление проектной идеи. Формирование программы работ. Введение в искусственный интеллект. Примеры на языке Python с искусственным интеллектом по отгадыванию чисел, метод дихотомии. Управление	6	2	4	

	искусственным интеллектом.				
8.3.	Подготовка к публичному выступлению для защиты результатов.	2	1	1	
8.4.	Демонстрация результатов работы	2		2	
9	Кейс «Игра Виселица»	12	3	9	
9.2.	Оформление проектной идеи. Формирование программы работ. Работа на языке Python со словарями и списками, множественное присваивание, добавление и удаление элементов в список.	2	1	1	
9.3.	Планирование дизайна и механики игры. Создание главного меню игры, подсчета очков.	4	1	3	
9.4.	Визуализирование программы в виде блок-схемы.	2	1	1	
9.5.	Тестирование написанной программы и доработка.	2		2	
9.6.	Подготовка к публичному выступлению для защиты результатов. Демонстрация результатов работы.	2		2	
10.	Продвинутое конструирование языка Python. Генерация на языке Python двумерного игрового поля, методики перетасовки элементов в массивах, игровой цикл.	2	1	1	
11.	Примеры на языке Python на темы: различия между словами и списками, обработка ошибочных предположений, игровое меню, завершение и перезагрузка игры.	4	2	2	
12.	Кейс «Игра Крестики-Нолики»	18	3	11	4
12.1.	Оформление проектной идеи. Формирование программы работ.	2	1	1	
12.2.	Текстовая версия с искусственным интеллектом, игра: человек с компьютером, компьютер с компьютером.	4	2	2	
12.3.	Обработка статистики, вывод и обработка результатов.	2		2	
12.4.	Тестирование написанной программы и доработка.	2		2	
12.5.	Оформление презентационной игры. 3D-печать.	4			4
12.6.	Подготовка к публичному выступлению для защиты результатов.	2		2	
12.7.	Демонстрация программы в группе и защита результатов.	2		2	
	ВСЕГО	72	25	43	4

Содержание учебно-тематического плана

а. Вводный модуль - линия 0 (IT-START 13-14 лет)

№	Темы занятия	Содержание занятий
1.	Вводное занятие. Введение в предмет, техника безопасности (2 ч)	Теория: Вводный инструктаж «Охрана труда на занятиях. Правила поведения на занятиях». Перспективы применения приобретённых знаний. Знакомство с оборудованием лаборатории. Правила противопожарной безопасности. Санитарно-гигиенические правила в соответствии с требованиями СанПиН 2.4.4.1251 – 03. Правила грамотного выполнения операций на оборудовании, правила пользования инструментами. Практика: Игры на сплочение и командообразование.
2.	Знакомство с операционной системой Linux Debian 9.6 (Или Аналогичной) (4 ч)	Теория: История зарождения операционных систем Unix/Linux. Отличия систем Unix/Linux от Windows систем. Устройство файловой системы в Unix/Linux. Практика: Установка Oracle VirtualBox. Установка Linux Debian 9.6 в качестве виртуальной машины. Знакомство с возможностями Linux.
3.	Освоение скриптового языка bash (ввод-вывод, архивирование, поиск файлов) (4 ч)	Теория: Программа Terminal в Linux. Возможности терминала. Понятие скриптов и команд. Понятие «программа». Понятие скриптового языка. Понятие интерпретатора. Практика: Запуск терминала. Выполнение команд в интерпретаторе. Вывод информации на экран, создание файлов и директорий, архивов, поиск информации в содержимых файлах. Сохранение и загрузка файлов программ.
4.	Освоение текстового редактора Vim (4 ч)	Теория: История текстовых редакторов. Отличия редактора Vim от всех остальных. Практика. Запуск и выход из Vim. Режимы команд, ввода и выделения. Интерактивные игры для самообучения навыкам набора текста в Vim.
5.	Изучение издательской системы TeX/LaTeX (4 ч)	Теория: Алгоритм создание PDF-файла. Сцепифика шаблона TeX-файла. Набор заголовков, формул, таблиц и картинок. Языковые типографические особенности

		букв и лигатур в научных статьях. Практика: Создание в редакторе Vim шаблона реферата и презентации. Компиляция из TeX в PDF.
6.	Написание и тестирование скриптов для администрирования системы Linux (4 ч)	Теория: Служебные команды в Linux. Управление дисковым пространством и установка сторонних программ. Добавление и удаление пользователей, контроль правами файлов и директорий пользователей. Практика: Написание и отладка сервисных скриптов, контроль за процессами в системе. Обработка скриптами полученной информации. Формирование служебных отчетов.
7.	Основы языка Python. Примеры на языке Python с разбором конструкций: циклы, условия, ветвления, массивы, типы данных. (3 ч)	Теория: История языка Python, сфера применения языка, различие в версиях, особенности синтаксиса. Объявление и использование переменных в Python. Использование строк, массивов, кортежей и словарей в Python. Использование условий, циклов и ветвлений в Python. Практика: Запуск интерпретатора из системы Linux. Различия интерпретатора и компилятора. Написание простейших демонстрационных программ. Мини-программы внутри программы. Выражения в вызовах функций. Имена переменных. Упражнения по написанию программ с использованием переменных, условий и циклов. Генерация случайных чисел. Группировка циклов в блоки. Операции сравнения.
8	Кейс «Числовые загадки»	
8.1.	Оформление проектной идеи. Формирование программы работ (1 ч)	Теория: Знакомство с кейсом, представление поставленной проблемы. Практика: Мозговой штурм. Анализ проблемы, генерация и обсуждение методов ее решения.
8.2.	Введение в искусственный интеллект. Примеры на языке Python с искусственным интеллектом по отгадыванию чисел,	Теория: Алгоритмы поиска числа в массиве. Варианты сортировок. Поиск дихотомией. Работа с переменными, работа с функциями. Практика: Упражнения по поиску чисел в массиве. Упражнения на сортировку чисел. Алгоритмы поиска числа. Исследование скорости работы алгоритмов.

	метод дихотомии. Управление искусственным интеллектом (6 ч)	
8.3.	Подготовка к публичному выступлению для защиты результатов (2 ч)	Теория: Создание удобной и понятной презентации. Практика: Подготовка презентации для защиты. Подготовка речи для защиты.
8.4.	Демонстрация отчета в группе и защита результатов работы (2 ч)	Практика: Презентация созданной программы.
9	Кейс «Игра Виселица»	
9.1.	Оформление проектной идеи. Формирование программы работ (1 ч)	Теория: Знакомство с кейсом, представление поставленной проблемы. Практика: Мозговой штурм. Анализ проблемы, генерация и обсуждение методов ее решения
9.2.	Работа на языке Python со словарями и списками, множественное присваивание, добавление и удаление элементов в список (4 ч)	Теория: Доступ к элементам по индексам. Получение слова из словаря. Отображение игрового поля игрока. Получение предположений игрока. Проверка допустимости предположений игрока. Практика: Создание прототипа программы. Отработка методик.
9.3.	Планирование дизайна и механики игры. Создание главного меню игры, подсчета очков. (2 ч)	Теория: Понятие «механика игры», ограничения, правила. Практика: Упражнения. Проверка наличия буквы в секретном слове. Проверка – не победил ли игрок. Обработка ошибочных предположений. Проверка – не проиграл ли игрок. Завершение или перезагрузка игры. Создание главного меню игры, реализация подсчета очков.
9.4.	Визуализирование программы в виде блок-схемы (2 ч)	Теория: Проектирование проекта с помощью блок-схем. Практика: Создание блок-схем. Ветвление в блок-схемах. Заканчиваем или начинаем игру сначала. Следующая попытка. Обратная связь с игроком.
9.5.	Тестирование написанной программы и доработка	Практика: Тестирование созданной игры программы, доработка и расширение возможностей.

	(1 ч)	
9.6.	Подготовка к публичному выступлению для защиты результатов (1 ч)	Практика: Подготовка презентации и речи для защиты.
9.7.	Демонстрация результатов работы (1 ч)	Практика: Презентация созданной программы.
10.	Продвинутые конструкции языка Python. Генерация на языке Python двумерного игрового поля, методики перетасовки элементов в массивах, игровой цикл (4 ч)	Теория: Система координат компьютерного экрана. Создание поля на новой карте. Изменение порядка элементов списка. Определение допустимости хода. Отражение хода на игровом поле. Практика: Упражнения на примере прототипа двумерной стратегической игры-заготовки.
11.	Примеры на языке Python на темы: различия между словами и списками, обработка ошибочных предположений, игровое меню, завершение и перезагрузка игры (6 ч)	Теория: Криптография и шифрование. Шифрование и расшифровывание сообщения. Выбор состояния программы из игрового меню. Практика: Пример программы «Шифр Цезаря».
12.	Кейс «Игра Крестики-Нолики»	
12.1.	Оформление проектной идеи. Формирование программы работ (1 ч)	Теория: Знакомство с кейсом, представление поставленной проблемы. Практика: Мозговой штурм. Анализ проблемы, генерация и обсуждение методов ее решения
12.2.	Текстовая версия с искусственным интеллектом, игра: человек с компьютером, компьютер с компьютером (2 ч)	Теория: Данные для прорисовки игрового поля. Стратегия игры искусственного интеллекта. Практика: Упражнения по написанию блоков игры: вывод игрового поля на экран, предоставлению игроку выбору кем ходить, размещение меток на игровом поле, дублирование данных игрового поля, вычисление по короткой схеме, создание искусственного интеллекта.
12.3.	Обработка статистики,	Практика: Создание главного меню игры, реализация

	вывод и обработка результатов (2 ч)	подсчета очков.
12.4.	Тестирование написанной программы и доработка (2 ч)	Практика: Тестирование созданной игры программы, доработка и расширение возможностей.
12.5.	Оформление презентационной игры. 3D-печать (4 ч)	Практика – работа в Хайтеке: Создание рекламной модели игры на 3D принтере.
12.6.	Подготовка к публичному выступлению для защиты результатов (2 ч)	Практика: Подготовка презентации и речи для защиты.
12.7.	Демонстрация результатов работы (1 ч)	Практика: Презентация созданной программы.

Содержание учебно-тематического плана

б. Вводный модуль - линия 0 (IT-TOTAL 15-17 лет)

№	Темы занятия	Содержание занятий
1.	Вводное занятие. Введение в предмет, техника безопасности (2 ч)	Теория: Вводный инструктаж «Охрана труда на занятиях. Правила поведения на занятиях». Перспективы применения приобретённых знаний. Знакомство с оборудованием лаборатории. Правила противопожарной безопасности. Санитарно-гигиенические правила в соответствии с требованиями СанПиН 2.4.4.1251 – 03. Правила грамотного выполнения операций на оборудовании, правила пользования инструментами. Практика: Игры на сплочение и командообразование.
2.	Знакомство с операционной системой Linux Debian 9.6 (Или Аналогичной) (4 ч)	Теория: История зарождения операционных систем Unix/Linux. Отличия систем Unix/Linux от Windows систем. Устройство файловой системы в Unix/Linux. Практика: Установка Oracle VirtualBox. Установка Linux Debian 9.6 в качестве виртуальной машины. Установка других Linux систем и сравнение их между собой. Знакомство с возможностями различных дистрибутивов Linux. Установка программ из репозитория и с сайтов разработчиков.

3.	Освоение скриптового языка bash (ввод-вывод, архивирование, поиск файлов) (4 ч)	Теория: Программа Terminal в Linux. Возможности терминала. Понятие скриптов и команд. Понятие «программа». Понятие скриптового языка. Понятие интерпретатора. Сравнение терминальных оболочек (bash, sh, tsh, и др.) Практика: Запуск терминала. Выполнение команд в интерпретаторе. Вывод информации на экран, создание файлов и директорий, поиск информации в содержимых файлах. Сохранение и загрузка файлов программ. Сравнение программ архивирования файлов. Современный стандарт запаковки и распаковки программ (.tar архив)
4.	Освоение текстового редактора Vim (4 ч)	Теория: История текстовых редакторов. Отличия редактора Vim от всех остальных. Практика. Запуск и выход из Vi. Установка Vim. Режимы команд, ввода и выделения. Интерактивные игры для самообучения навыкам набора текста в Vim. Плагины и расширения для Vim. Компиляция текста программы из Vim. Связь редактора Vim с языками программирования.
5.	Изучение издательской системы TeX/LaTeX (4 ч)	Теория: Алгоритм создания PDF-файла. Специфика шаблона TeX-файла. Набор заголовков, формул, таблиц и картинок. Языковые типографические особенности букв и лигатур в научных статьях. Практика: Создание в редакторе Vim шаблонов научной статьи, реферата и презентации. Компиляция из TeX в PDF. Специфика уровней заголовков и библиографии (списка литературы). Подключение стилей (русского и английского).
6.	Написание и тестирование скриптов для администрирования системы Linux (4 ч)	Теория: Служебные команды в Linux. Управление дисковым пространством и установка сторонних программ. Добавление и удаление пользователей, контроль правами файлов и директорий пользователей. Практика: Написание и отладка сервисных скриптов, контроль за процессами в системе. Обработка скриптами полученной информации. Формирование служебных отчетов. Подключение к удаленному

		терминалу на другом сервере через зашифрованный ssh-канал.
7.	Основы языка Python. Примеры на языке Python с разбором конструкций: циклы, условия, ветвления, массивы, типы данных (4 ч)	Теория: История языка Python, сфера применения языка, различие в версиях, особенности синтаксиса. Объявление и использование переменных в Python. Использование строк, массивов, кортежей и словарей в Python. Использование условий, циклов и ветвлений в Python. Практика: Запуск интерпретатора из системы Linux. Различия интерпретатора и компилятора. Написание простейших демонстрационных программ. Мини-программы внутри программы. Выражения в вызовах функций. Имена переменных. Упражнения по написанию программ с использованием переменных, условий и циклов. Генерация случайных чисел. Группировка циклов в блоки. Операции сравнения. Бесконечные циклы с различными условиями выхода из них.
8.	Кейс «Числовые загадки»	
8.1.	Оформление проектной идеи. Формирование программы работ (1 ч)	Теория: Знакомство с кейсом, представление поставленной проблемы. Практика: Мозговой штурм. Анализ проблемы, генерация и обсуждение методов ее решения.
8.2.	Введение в искусственный интеллект. Примеры на языке Python с искусственным интеллектом по отгадыванию чисел, метод дихотомии. Управление искусственным интеллектом (6 ч)	Теория: Алгоритмы поиска числа в массиве. Варианты сортировок. Поиск дихотомией. Работа с переменными, работа с функциями. Практика: Упражнения по поиску чисел в массиве. Упражнения на сортировку чисел. Алгоритмы поиска числа. Исследование скорости работы алгоритмов. Преобразование текстовых данных в графики на Python через библиотеку matplotlib. Разбор элементов графика, управление цветом, опциями и свойствами элементов.
8.3.	Подготовка к публичному выступлению для защиты	Теория: Как проходят публичные выступления. Как держать себя на сцене. Создание удобной и понятной

	результатов (2 ч)	презентации. Практика: Подготовка презентации для защиты. Подготовка речи для защиты.
8.4.	Демонстрация отчета в группе и защита результатов работы (1 ч)	Практика: Презентация созданной программы.
9.	Кейс «Игра Виселица»	
9.1.	Оформление проектной идеи. Формирование программы работ (1 ч)	Теория: Знакомство с кейсом, представление поставленной проблемы. Практика: Мозговой штурм. Анализ проблемы, генерация и обсуждение методов ее решения.
9.2.	Работа на языке Python со словарями и списками, множественное присваивание, добавление и удаление элементов в список (2 ч)	Теория: Доступ к элементам по индексам. Получение слова из словаря. Отображение игрового поля игрока. Получение предположений игрока. Проверка допустимости предположений игрока. Практика: Создание прототипа программы. Варианты получения секретного слова из списка. Отработка методик.
9.3.	Планирование дизайна и механики игры. Создание главного меню игры, подсчета очков (3 ч)	Теория: Понятие «механика игры», ограничения, правила. Практика: Понятие «механика игры», ограничения, правила. Понятие игрового цикла. Упражнения. Проверка наличия буквы в секретном слове. Проверка – не победил ли игрок. Обработка ошибочных предположений. Проверка – не проиграл ли игрок. Завершение или перезагрузка игры. Создание главного меню игры, реализация подсчета очков. Накопление и вывод игровой статистики.
9.4.	Визуализирование программы в виде блок-схемы (2 ч)	Теория: Проектирование проекта с помощью блок-схем. Практика: Проектирование проекта с помощью блок-схем. Создание блок-схем. Ветвление в блок-схемах. Заканчиваем или начинаем игру сначала. Следующая попытка. Обратная связь с игроком.
9.5.	Тестирование написанной программы и доработка (2 ч)	Практика: Тестирование созданной игры программы, доработка и расширение возможностей.

9.6.	Подготовка к публичному выступлению для защиты результатов (1 ч)	Практика: Подготовка презентации и речи для защиты.
9.7.	Демонстрация результатов работы (2 ч)	Практика: Презентация созданной программы
10.	Продвинутые конструкции языка Python. Генерация на языке Python двумерного игрового поля, методики перетасовка элементов в массивах, игровой цикл (2 ч)	Теория: Система координат компьютерного экрана. Создание поля на новой карте. Изменение порядка элементов списка. Определение допустимости хода. Отражение хода на игровом поле. Практика: Упражнения на примере прототипа двумерной стратегической игры-заготовки. Демонстрация игроку статуса игры. Обработка хода игрока. Проверка победы игрока. Проверка проигрыша игрока. Завершение работы программы.
11.	Примеры на языке Python на темы: различия между словами и списками, обработка ошибочных предположений, игровое меню, завершение и перезагрузка игры (4 ч)	Теория: Криптография и шифрование. Шифрование и расшифровывание сообщения. Выбор состояния программы из игрового меню. Практика: Пример программы «Шифр Цезаря». Шифрование/расшифровка каждой буквы. Полный перебор. Получение ключа от игрока. Управление длиной ключа.
12.	Кейс «Игра Крестики-Нолики»	
12.1.	Оформление проектной идеи. Формирование программы работ (2 ч)	Теория: Знакомство с кейсом, представление поставленной проблемы. Практика: Мозговой штурм. Анализ проблемы, генерация и обсуждение методов ее решения
12.2	Текстовая версия с искусственным интеллектом, игра: человек с компьютером, компьютер с компьютером (4 ч)	Теория: Данные для прорисовки игрового поля. Стратегия игры искусственного интеллекта. Практика: Упражнения по написанию блоков игры: вывод игрового поля на экран, предоставлению игроку выбору кем ходить, размещение меток на игровом поле, дублирование данных игрового поля, вычисление по короткой схеме, создание искусственного интеллекта. Управление игровым циклом. Выбор лучшего хода из списка. Проверка угловых, центральной и боковой клеток.

12.3.	Обработка статистики, вывод и обработка результатов (2 ч)	Практика: Создание главного меню игры, выбор сложности игры, реализация подсчета очков. Сражение искусственных интеллектов компьютер-компьютер.
12.4.	Тестирование написанной программы и доработка (2 ч)	Практика: Тестирование созданной игры программы, доработка и расширение возможностей.
12.5.	Оформление презентационной игры. 3D-печать (4 ч)	Практика – работа в Хайтеке: Создание рекламной модели игры на 3D принтере.
12.6.	Подготовка к публичному выступлению для защиты результатов (2 ч)	Практика: Подготовка презентации и речи для защиты.
6.6.	Демонстрация результатов работы (2 ч)	Практика: Презентация созданной программы.

Базовый модуль - линия 1

Базовым форматом образовательного процесса является проектная деятельность. Школьник, будучи вовлеченным в проектную деятельность и имея в голове образ конечного результата, мотивирован на «добычу» знаний, у него формируется потребность в приобретении знаний.

Гибкость образовательных траекторий обеспечивается предоставлением школьнику возможности самостоятельно выбирать роль и степень участия в проекте.

Каждой из возрастных групп предлагается по два одинаковых кейса, но с различным уровнем сложности выполнения.

Темы кейсов для проектной деятельности: «Игра Реверси», «Игра Ловкач», а также работа над федеральными кейсами и подготовка к конкурсам.

Учебно-тематический план
Базовый модуль - линия 1 для возрастов: IT-START 13-14 лет и
IT-TOTAL 15-17 лет

№ п/п	Наименование разделов, тем	Всего	Проектные часы	
			Теория	Практикум
1	Предпроектный этап	11	4	7
	1.1. Введение в контекст	2	1	1
	1.2. Постановка проблемы	2	1	1
	1.3. Дорожная карта проекта	3	1	2
	1.4. Подбор и освоение алгоритмов исследования /проектирования	4	1	3
2	Работа в проекте	17		17
	2.1. Оформление проектной идеи	4		4
	2.2. Движение по дорожной карте проекта	13		13
3	Отчетный этап: оформление проектной идеи	8	1.5	6.5
	3.1. Навыки ораторского мастерства	2	0.5	1.5
	3.2. Подготовка презентации к защите проекта	3	1	2
	3.3. Защита проекта	1		1
	3.4. Проектирование шага развития	2		2
Всего:		36	5,5	30,5

Содержание учебно-тематического плана

Базовый модуль - линия 1

а. Кейс «Игра Реверси». Возраст: IT-START 13-14 лет

№	Темы занятия	Содержание занятий
1	Предпроектный этап	
1.1.	Введение в контекст (2 ч)	Теория: Встреча-знакомство с проектной группой, определение круга вопросов по реализации проекта. Формирование умения договариваться и решать совместно поставленную задачу (1 ч) Практика: Обзор достижений современных

		технологий. Определение связи конкретных изобретений и технологий с научными открытиями, перечень нерешенных задач, обсуждаемых экспертными сообществами (1 ч)
1.2.	Постановка проблемы (2 ч)	Теория: Правила проведения дискуссии, разбор проблемных ситуаций проекта с выявлением противоречий, их формулировка. Методы генерации новых идей (1 ч) Практика: Дискуссионный практикум (отработка коммуникационных навыков обучающихся): - определение наиболее интересующей проектную группу проблемы - определение, в какой форме можно реализовать решение - определение дополнительных предметных и технических знаний, необходимых для решения проблемы (1 ч)
1.3.	Дорожная карта проекта (3 ч)	Теория: Знакомство с дорожной картой IT- проекта, проектной технологией Scrum, понятием «заказчик» (1 ч) Практика: Работа с картой. Выбор темы проекта, личное осознание актуальности темы, определение, как и кем будет использоваться результат проектной деятельности. Оценка возможностей. Определение задач и их описание (2 ч)
1.4.	Подбор и освоение алгоритмов исследования/проектирования (4 ч)	Теория: Работа с источниками (1 ч) Практика: Самостоятельная работа с информацией: основные сведения о современном состоянии выбранной технологической сферы, об инвариантных технологических решениях выбранной сферы. Беседа, мозговой штурм: обсуждение технологий, использование которых могло бы быть уместно при решении проблемы. Как они будут работать и как они могут быть полезны для вашего проекта. Разработка плана ситуационного анализа (3 ч)
2	Работа в проекте	
2.1	Оформление проектной идеи	Практика: Презентация намерений группы:

	(4 ч)	описание проблемной ситуации с учетом полученных знаний. Работа с дорожной картой IT-проекта. Формирование задач, способствующих решению проектной задачи. Выделение основных ограничений. Формирование идеального образа решения (4 ч)
2.3	Движение по дорожной карте проекта (13 ч) ✓ Планируем ✓ Разрабатываем и создаем ✓ Тестируем ✓ Дорабатываем ✓ Обсуждаем	Практика: Индивидуальная работа над задачами и подзадачами проекта. Структура данных игрового поля. Проверка допустимости хода. Проверка допустимости координат. Получение игрового счета. Определение первого игрока. Помещение фишки на поле. Создание копии структуры данных игрового поля. Получение хода игрока. Вывод игрового счета на экран. Игровой цикл. (13 ч)
3	Отчетный этап	
3.1.	Навыки ораторского мастерства (2 ч)	Теория: Навыки ораторского мастерства: работа над дикцией и интонацией. Как эффективно справиться со страхом перед выступлением (0,5 ч) Практика: упражнения на развития голоса, дикции и выразительности речи (1,5 ч)
3.2.	Подготовка презентации к защите проекта (3 ч)	Теория: Формат и структура презентации (1 ч) Правила успешной защиты проекта Практика: Самостоятельная работа над презентацией проекта: демонстрация и обоснование схемы решения, находки, действующее решение, как может быть применена полезная функция, в чем ее полезность, какие интересные решения были найдены во время работы (2 ч)
3.3.	Защита проекта (1 ч)	Практика: Публичная демонстрация системы и защита презентации (1 ч)
3.4.	Проектирование шага развития (2 ч)	Практика: Рефлексия. Выявление непредусмотренных эффектов. Совместное решение о смене темы проекта или продолжения работы над его совершенствованием. Определение - в каких областях науки и техники стоит углубить свои знания и получить дополнительную практику, и с помощью

		педагога сформировать индивидуальную образовательную траекторию (2 ч)
--	--	---

б. Кейс «Игра Ловкач». Возраст: IT-START 13-14 лет

№	Темы занятия	Содержание занятий
1	Предпроектный этап	
1.1.	Введение в контент (2 ч)	Теория: Встреча-знакомство с проектной группой, определение круга вопросов по реализации проекта. Формирование умения договариваться и решать совместно поставленную задачу (1 ч) Практика: Обзор достижений современных технологий. Определение связи конкретных изобретений и технологий с научными открытиями, перечень нерешенных задач обсуждаемых экспертными сообществами (1 ч)
1.2.	Постановка проблемы (2 ч)	Теория: Правила проведения дискуссии, разбор проблемных ситуаций проекта с выявлением противоречий, их формулировка. Методы генерации новых идей (1 ч) Практика: Дискуссионный практикум (отработка коммуникационных навыков обучающихся): - определение наиболее интересующей проектную группу проблемы) - определение, в какой форме можно реализовать решение - определение дополнительных предметных и технических знаний необходимых для решения проблемы (1 ч)
1.3.	Дорожная карта проекта (3 ч)	Теория: Знакомство с дорожной картой IT-проекта, проектной технологией Scrum, понятием заказчик (1 ч) Практика: Работа с картой. Выбор темы проекта, личное осознание актуальности темы, определение, как и кем, будет использоваться результат проектной деятельности. Оценка возможностей. Определение задач и их описание (2 ч)
1.4.	Подбор и освоение алгоритмов исследования/проектирования (4 ч)	Теория: Работа с источниками (1 ч) Практика: Самостоятельная работа с информацией: основные сведения о современном

		состоянии выбранной технологической сферы, об инвариантных технологических решениях выбранной сферы. Беседа, мозговой штурм: обсуждение технологий, использование которых могло бы быть уместно при решении проблемы, как они будут работать и как они могут быть полезны для вашего проекта. Разработка плана ситуационного анализа (3 ч)
2	Работа в проекте	
3.1	Оформление проектной идеи (4 ч)	Практика: Презентация намерений группы: описание проблемной ситуации с учетом полученных знаний. Работа с дорожной картой IT-проекта. Формирование задач, способствующих решению проектной задачи. Выделение основных ограничений. Формирование идеального образа решения (4 ч)

3.2	Движение по дорожной карте проекта (13 ч) ✓ Планируем ✓ Разрабатываем и создаем ✓ Тестируем ✓ Дорабатываем ✓ Обсуждаем	Практика: Индивидуальная работа над задачами и подзадачами проекта. Создание собственной мини-библиотеки Python. Рисование графики в Python. Графическая библиотека PyGame. Рисование спрайтов на языке Python. Создание графического окна с перемещающимися спрайтами. Разработка прототипа 2D стратегии Составляющие части игры - стратегии. Список объектов. Игровое поле как двумерный массив. Вечный цикл. Создание прототипа 2D стратегии на языке Python Разработка искусственного интеллекта. Алгоритмы принятия решений. Базовое восприятие. Системы на основе правил. Конечные автоматы. Создание простой версии искусственного интеллекта для стратегической игры. Расширение искусственного интеллекта. Усовершенствование искусственного интеллекта для стратегической игры. Тестирование написанной программы и доработка. Запуск программы с валидными и невалидными параметрами. Подготовка к соревнованию искусственных интеллектов. Составление турнирной таблицы. Подготовка системы проведения соревнований (13 ч)
4	Отчетный этап	
4.1.	Навыки ораторского мастерства (2 ч)	Теория: Навыки ораторского мастерства: работа над дикцией и интонацией. Как эффективно справляться с страхом перед выступлением (0,5 ч) Практика: упражнения на развития голоса, дикции и выразительности речи (1,5 ч)
4.2.	Подготовка презентации к защите проекта (3 ч)	Теория: Формат и структура презентации. Правила успешной защиты проекта (1 ч) Практика: Самостоятельная работа над презентацией проекта: демонстрация и обоснование схемы решения, находки,

		действующее решение, как может быть применена полезная функция, в чем ее полезность, какие интересные решения были найдены во время работы (2 ч)
4.3.	Защита проекта (1 ч)	Практика: Публичная демонстрация системы и защита презентации (1 ч)
4.4.	Проектирование шага развития (2 ч)	Практика: Рефлексия. Выявление непредусмотренных эффектов. Совместное решение о смене темы проекта или продолжения работы над его совершенствованием. Определение в каких областях науки и техники стоит углубить свои знания и получить дополнительную практику, и с помощью педагога сформировать индивидуальную образовательную траекторию (2 ч)

с. Кейс «Игра Реверси». Возраст: IT-TOTAL 15-17 лет

№	Темы занятия	Содержание занятий
1	Предпроектный этап	
1.1.	Введение в контекст (4 ч)	Теория: Встреча-знакомство с проектной группой, определение круга вопросов по реализации проекта. Формирование умения договариваться и решать совместно поставленную задачу (2 ч) Практика: Обзор достижений современных технологий. Определение связи конкретных изобретений и технологий с научными открытиями, перечень нерешенных задач, обсуждаемых экспертными сообществами (2 ч)
1.2.	Постановка проблемы (4 ч)	Теория: Правила проведения дискуссии, разбор проблемных ситуаций проекта с выявлением противоречий, их формулировка. Методы генерации новых идей (2 ч) Практика: Дискуссионный практикум (отработка коммуникационных навыков обучающихся): - определение наиболее интересующей проектную группу проблемы - определение, в какой форме можно реализовать решение - определение дополнительных предметных и технических знаний, необходимых для решения проблемы (2 ч)
1.3.	Дорожная карта проекта (6 ч)	Теория: Знакомство с дорожной картой IT- проекта, проектной технологией Scrum, понятием «заказчик» (2 ч) Практика: Работа с картой. Выбор темы проекта, личное осознание актуальности темы, определение, как и кем будет использоваться результат проектной деятельности. Оценка возможностей. Определение задач и их описание (4 ч)
1.4.	Подбор и освоение алгоритмов исследования/проектирования (8 ч)	Теория: Работа с источниками (2 ч) Практика: Самостоятельная работа с информацией: основные сведения о современном состоянии

		выбранной технологической сферы, об инвариантных технологических решениях выбранной сферы. Беседа, мозговой штурм: обсуждение технологий, использование которых могло бы быть уместно при решении проблемы. Как они будут работать и как они могут быть полезны для вашего проекта. Разработка плана ситуационного анализа (6 ч)
2	Работа в проекте	
2.1	Оформление проектной идеи (8 ч)	Практика: Презентация намерений группы: описание проблемной ситуации с учетом полученных знаний. Работа с дорожной картой IT-проекта. Формирование задач, способствующих решению проектной задачи. Выделение основных ограничений. Формирование идеального образа решения (8 ч)
2.3	Движение по дорожной карте проекта (26 ч) ✓ Планируем ✓ Разрабатываем и создаем ✓ Тестируем ✓ Дорабатываем ✓ Обсуждаем	Практика: Индивидуальная работа над задачами и подзадачами проекта. Практика: Индивидуальная работа над задачами и подзадачами проекта. Структура данных игрового поля. Проверка допустимости хода. Проверка допустимости координат. Получение игрового счета. Определение первого игрока. Помещение фишки на поле. Создание копии структуры данных игрового поля. Получение хода игрока. Вывод игрового счета на экран. Игровой цикл. Компьютер против компьютера. Удаление приглашений для игрока и добавление игрока-компьютера. Отслеживание результатов партий. Оценка искусственных интеллектов в процентном соотношении. Сравнение различных алгоритмов ИИ. ИИ лучшего углового хода. ИИ худшего хода. ИИ случайного хода. ИИ худшего хода против ИИ лучшего хода. ИИ случайного хода против ИИ лучшего игрового хода. ИИ лучшего углового-граничного хода против ИИ лучшего игрового хода. (18 ч)
3	Отчетный этап	

3.1.	Навыки ораторского мастерства (4 ч)	Теория: Навыки ораторского мастерства: работа над дикцией и интонацией. Как эффективно справляться со страхом перед выступлением (1 ч) Практика: упражнения на развития голоса, дикции и выразительности речи (3 ч)
3.2.	Подготовка презентации к защите проекта (6 ч)	Теория: Формат и структура презентации (2 ч) Правила успешной защиты проекта Практика: Самостоятельная работа над презентацией проекта: демонстрация и обоснование схемы решения, находки, действующее решение, как может быть применена полезная функция, в чем ее полезность, какие интересные решения были найдены во время работы (4 ч)
3.3.	Защита проекта (2 ч)	Практика: Публичная демонстрация системы и защита презентации (2 ч)
3.4.	Проектирование шага развития (4 ч)	Практика: Рефлексия. Выявление непредусмотренных эффектов. Совместное решение о смене темы проекта или продолжения работы над его совершенствованием. Определение - в каких областях науки и техники стоит углубить свои знания и получить дополнительную практику, и с помощью педагога сформировать индивидуальную образовательную траекторию (4 ч)

d. Кейс «Игра Ловкач». Возраст: IT-TOTAL 15-17 лет

№	Темы занятия	Содержание занятий
1	Предпроектный этап	
1.1.	Введение в контент (4 ч)	Теория: Встреча-знакомство с проектной группой, определение круга вопросов по реализации проекта. Формирование умения договариваться и решать совместно поставленную задачу (2 ч) Практика: Обзор достижений современных технологий. Определение связи конкретных изобретений и технологий с научными открытиями, перечень нерешенных задач обсуждаемых экспертными сообществами (2 ч)
1.2.	Постановка проблемы (4 ч)	Теория: Правила проведения дискуссии, разбор проблемных ситуаций проекта с выявлением противоречий, их формулировка. Методы генерации новых идей (2 ч) Практика: Дискуссионный практикум (отработка коммуникационных навыков обучающихся): - определение наиболее интересующей проектную группу проблемы) - определение, в какой форме можно реализовать решение - определение дополнительных предметных и технических знаний необходимых для решения проблемы (2 ч)
1.3.	Дорожная карта проекта (6 ч)	Теория: Знакомство с дорожной картой IT-проекта, проектной технологией Scrum, понятием заказчик (2 ч) Практика: Работа с картой. Выбор темы проекта, личное осознание актуальности темы, определение, как и кем, будет использоваться результат проектной деятельности. Оценка возможностей. Определение задач и их описание (4 ч)
1.4.	Подбор и освоение алгоритмов исследования/проектирования (8 ч)	Теория: Работа с источниками (2 ч) Практика: Самостоятельная работа с информацией: основные сведения о современном

		состоянии выбранной технологической сферы, об инвариантных технологических решениях выбранной сферы. Беседа, мозговой штурм: обсуждение технологий, использование которых могло бы быть уместно при решении проблемы, как они будут работать и как они могут быть полезны для вашего проекта. Разработка плана ситуационного анализа (6 ч)
2	Работа в проекте	
3.1	Оформление проектной идеи (8 ч)	Практика: Презентация намерений группы: описание проблемной ситуации с учетом полученных знаний. Работа с дорожной картой IT-проекта. Формирование задач, способствующих решению проектной задачи. Выделение основных ограничений. Формирование идеального образа решения (8 ч)

3.2	Движение по дорожной карте проекта (26 ч) ✓ Планируем ✓ Разрабатываем и создаем ✓ Тестируем ✓ Дорабатываем ✓ Обсуждаем	Практика: Индивидуальная работа над задачами и подзадачами проекта. Введение в язык Python. Создание собственной мини-библиотеки Python. Рисование графики в Python. Графическая библиотека PyGame. Работа с цветом. Вывод текста в окне. Заливка цветом объекта. Окрашивание пикселей. Рисование спрайтов на языке Python. Создание графического окна с перемещающимися спрайтами. Разработка прототипа 2D стратегии (4 ч) Составляющие части игры - стратегии. Список объектов. Игровое поле как двумерный массив. Вечный цикл. Создание прототипа 2D стратегии на языке Python (4 ч) Анимированная графика. Обнаружение столкновений. Использование звуков и изображений. Разработка искусственного интеллекта. Алгоритмы принятия решений. Базовое восприятие. Системы на основе правил. Конечные автоматы. Создание простой версии искусственного интеллекта для стратегической игры. Расширение искусственного интеллекта. Усовершенствование искусственного интеллекта для стратегической игры. Реализация чит-кодов. Измерение игры. Тестирование написанной программы и доработка. Запуск программы с валидными и невалидными параметрами. Подготовка к соревнованию искусственных интеллектов. Составление турнирной таблицы. Подготовка системы проведения соревнований (18 ч)
4	Отчетный этап	
4.1.	Навыки ораторского мастерства (4 ч)	Теория: Навыки ораторского мастерства: работа над дикцией и интонацией. Как эффективно справляться с страхом перед выступлением (1 ч) Практика: упражнения на развития голоса, дикции

		и выразительности речи (3 ч)
4.2.	Подготовка презентации к защите проекта (6 ч)	Теория: Формат и структура презентации. Правила успешной защиты проекта (2 ч) Практика: Самостоятельная работа над презентацией проекта: демонстрация и обоснование схемы решения, находки, действующее решение, как может быть применена полезная функция, в чем ее полезность, какие интересные решения были найдены во время работы (4 ч)
4.3.	Защита проекта (2 ч)	Практика: Публичная демонстрация системы и защита презентации (2 ч)
4.4.	Проектирование шага развития (4 ч)	Практика: Рефлексия. Выявление непредусмотренных эффектов. Совместное решение о смене темы проекта или продолжения работы над его совершенствованием. Определение в каких областях науки и техники стоит углубить свои знания и получить дополнительную практику, и с помощью педагога сформировать индивидуальную образовательную траекторию (4 ч)

Учебно-методическое обеспечение

Организационно-педагогические основы программы

Дополнительная общеобразовательная общеразвивающая программа «ИТ-Квантум» рассчитана на школьников в возрасте от 13 до 17 лет. При наборе детей в группы принимаются все желающие, на первых занятиях проводится собеседование с целью выявления уровня компьютерной грамотности.

Формы организации образовательного процесса

Формы проведения занятий: лекция, объяснение материала с привлечением обучающихся, самостоятельная исследовательская работа, эвристическая беседа, практическое учебное занятие, самостоятельная работа, проектная деятельность.

Формы организации учебной деятельности: индивидуальная (обучающемуся дается самостоятельное задание с учетом его возможностей), фронтальная (работа со всеми одновременно, например, при объяснении нового материала или отработке определённого технологического приёма), групповая (разделение обучающихся на группы для выполнения определённой работы).

Современные педагогические технологии, такие как: технология проектного обучения, ТРИЗ технологии, здоровьесберегающие технологии и другие в сочетании с современными информационными технологиями могут существенно повысить эффективность образовательного процесса, решить стоящие перед наставником задачи воспитания всесторонне развитой, творчески свободной личности.

В конце каждого занятия подводятся итоги, строятся планы на следующие занятия.

Основные методы обучения

Основной метод: проектный, кейсовый.

Метод эвристических вопросов предполагает для отыскания сведений о каком-либо событии или объекте задавать следующие семь ключевых вопросов: Кто? Что? Зачем? Чем? Где? Когда? Как?

Метод сравнения применяется для сравнения разных версий моделей обучающихся с созданными аналогами.

Метод эвристического наблюдения ставит целью научить детей добывать и конструировать знания с помощью наблюдений. Одновременно с получением заданной педагогом информации многие обучающиеся видят и другие особенности объекта, т.е. добывают новую информацию и конструируют новые знания.

Метод фактов учит отличать то, что видят, слышат, чувствуют обучающиеся, от того, что они думают.

Метод конструирования понятий начинается с актуализации уже имеющихся представлений обучающихся.

Метод прогнозирования применяется к реальному или планируемому процессу. Спустя заданное время прогноз сравнивается с реальностью. Проводится обсуждение результатов, делаются выводы.

Метод ошибок предполагает изменение устоявшегося негативного отношения к ошибкам, замену его на конструктивное использование ошибок. Ошибка рассматривается как источник противоречий, феноменов, исключений из правил, новых знаний, которые рождаются на противопоставлении общепринятым.

Креативные методы обучения ориентированы на создание обучающимися личного образовательного продукта путем накопленных знаний и поиском оптимального решения проблемы.

«Мозговой штурм» ставит основной задачей сбор как можно большего числа идей в результате освобождения участников обсуждения от инерции мышления и стереотипов.

Метод планирования предполагает планирование образовательной деятельности на определенный период - занятие, неделю, тему, творческую работу.

Метод рефлексии помогает обучающимся формулировать способы своей деятельности, возникающие проблемы, пути их решения и полученные

результаты, что приводит к осознанному образовательному процессу.

Метод самооценки вытекают из методов рефлексии, носят количественный и качественный характер, отражают полноту достижения обучающимся цели.

Создание ситуаций успеха на занятиях является одним из основных методов эмоционального стимулирования и представляет собой специально созданные педагогом цепочки таких ситуаций, в которых обучающийся добивается хороших результатов, что ведёт к возникновению у него чувства уверенности в своих силах и «лёгкости» процесса обучения.

Материально-техническое обеспечение

- ✓ Персональный компьютер с операционной системой Windows 10 и/или Debian Linux 9.6 (или аналогичной RedHat Linux, Ubuntu Linux и др.)
- ✓ Доступ в интернет
- ✓ Офисный пакет ПО Microsoft Office или аналогичный (OpenOffice, LibreOffice и др.)
- ✓ Установленное программное обеспечение TeX/LaTeX или аналогичный (MicTeX, TeXLive и др.)
- ✓ Установленное программное обеспечение Python 3.5 с фреймворком matplotlib

Список рекомендуемой литературы

Нормативно-правовые документы

- ✓ Федеральный закон от 29.12.12 г. №273-ФЗ «Об образовании в Российской Федерации»;
- ✓ Приказ Минпросвещения России от 09.11.18 №196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» (Зарегистрировано в Минюсте РФ 29.11.18 №52831);
- ✓ Концепция развития дополнительного образования детей в Российской Федерации, утвержденной распоряжением Правительства РФ от 4.09.2014г. № 1726-р;
- ✓ Санитарно-эпидемиологические правила и нормативы 2.4.4.3172-14
- ✓ «Требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей (утв. Главным государственным санитарным врачом РФ от 04.07.2014 г. № 41);
- ✓ Государственная программа РФ «Развитие образования на 2013-2020 годы, утвержденной постановлением Правительства РФ от 15.04.2014 г. № 295;
- ✓ Стратегия инновационного развития Российской Федерации на период до 2020 года, утвержденным распоряжением Правительства РФ от 08.12.2011 года № 2227-р;
- ✓ Федеральная целевая программа развития образования на 2016-2020 годы, утвержденной Постановлением Правительства РФ от 23.05.2015 года № 497.

Список используемой литературы:

Для педагогов:

- ✓ Андреева Е.В., Босова Л.Л., Фалина И.Н. Математические основы информатики. Элективный курс: Учебное пособие. – М.: БИНОМ. Лаборатория Знаний, 2007.
- ✓ Босова Л.Л., Босова А.Ю., Коломенская Ю.Г. Занимательные задачи по

информатике. – М.: БИНОМ. Лаборатория знаний. 2007.

- ✓ Долинский М.С. Алгоритмизация и программирование на Turbo Pascal: от простых до олимпиадных задач: Учебное пособие. – СПб.: Питер Принт, 2004.
- ✓ Долинский М.С. Решение сложных и олимпиадных задач по программированию. – СПб.: Питер, 2006.
- ✓ Златопольский Д. М. Программирование: типовые задачи, алгоритмы, методы. – М.: БИНОМ. Лаборатория знаний, 2007.

**Министерство образования, науки и молодежной политики
Нижегородской области
Государственное бюджетное учреждение дополнительного
образования «Центр молодежных инженерных и научных компетенций
«КВАНТОРИУМ»**

Дорожная карта проекта «_____»

Срок реализации: _____

Пояснительная записка

Данная проектная деятельность представляет собой образовательный маршрут для обучающихся

Куратор проекта: _____

Наставник: _____ педагог ДО

Участники проектной деятельности:

1. _____
2. _____
3. _____
4. _____
5. _____

Актуальность:

Особенности реализации:

Цель проекта:

Задачи:

- _____
- _____
- _____

Особенности организации образовательного процесса по индивидуальному плану: занятия проводятся 1 раз в неделю по 1,5 часа, в среду 18.00 – 19.15

Предполагаемые результаты:

Содержание программы:**Сферы ответственности наставника и наставляемого**

Общая сфера ответственности	Сфера ответственности наставника	Сфера ответственности обучающихся
1. Совместное планирование реализации проекта 2. Совместное обсуждение и поддержка обратной связи.	1. Ориентация на потребности и возможности участников проекта. 2. Конструктивная оценка подготовки: анализ знаний на основании вопросов, ошибок. 3. Предварительная подготовка материалов, идей и предложений. 4. Оптимизация времени обучения – он-лайн консультации.	1. Предварительная подготовка к встречам: вопросы по тематике проекта. 2. Внесение предложений по улучшению процесса проектной деятельности. 3. Всесторонняя помощь в организации общего процесса.

Формы обучения: групповая работа и индивидуальная работа, проектные практикумы, экскурсии.

Возможное количество обучающихся – до 5 человек в одной группе.

Сроки:

Начало _____

Окончание _____

План работы (укрупненный)

№	Задача	Срок
1		

2		
3		
4		
5		
6		
7		

План работы на период с _____ по _____

Планируемая дата выполнения задачи	Фактическая дата выполнения задачи	Задачи

Дата _____

Задачи к выполнению в течение занятия (план на занятие)

Дата _____

Результат работы _____

Указать причины отклонения от плана занятия _____

Планируемые задачи на следующее занятие

Проектная группа Наставник проекта

Кейс «Числовые загадки».

Описание проблемной ситуации.

Компьютер загадывает секретное число от 0 до 100 и просит пользователя угадать это число. После каждой попытки угадать, компьютер будет сообщать пользователю, было его число больше или меньше загаданного. Пользователь выиграет, если угадает число за шесть ходов.

Программа затрагивает много ключевых моментов программирования: конвертирования типов данных, запись и чтение файлов, использование алгоритма деления отрезка пополам, обработку полученных данных и представление их в виде графиков.

Категория кейса: Вводный.

Ход работы:

Занятие 1		Занятие 2		Занятие 3	
Цель: Постановка проблемы, генерация путей решения		Цель: Разработка алгоритма поиска числа методом деления отрезка пополам		Цель: Представление данных в виде графиков в Python	
Что делается: Представление поставленной проблемы группе детей. Анализ проблемы, генерация и обсуждение методов ее решения.	Компетенции: Умение искать информацию в различных источниках. Умение генерировать идеи предложенными методами.	Что делается: Упражнения по поиску чисел в массиве. Упражнения на сортировку чисел. Алгоритмы поиска числа. Исследование скорости работы алгоритмов.	Компетенции: Умение программировать на языке Python.	Что делается: Преобразование текстовых данных в графики на Python через библиотеку matplotlib. Разбор элементов графика, управление цветом, опциями и свойствами элементов.	Компетенции: Умение программировать на языке Python.
Занятие 4		Занятие 5			
Цель: Подготовка к публичному выступлению для защиты результатов		Цель: Демонстрация игры в группе и защита результатов			

Что делается: Подготовка речи и презентации в TeX/LaTeX для публичной демонстрации результатов работы в кейсе.	Компетенции: Умение анализировать результаты работы. Грамотное представление результатов своей деятельности.	Что делается: Публичная презентация результатов работы. Ответы на вопросы жюри. Рефлексия.	Компетенции: Базовые навыки ораторства, публичных выступлений, аргументирования точки зрения.		
--	---	--	---	--	--

Место кейса в структуре модуля: Базовый, мотивационный кейс.

Предполагаемые образовательные результаты учащихся (артефакты, решения), формируемые навыки (soft и hard skills):

- ✓ Развитие аналитического и алгоритмического мышления;
- ✓ Понятие алгоритма, последовательного выполнения действий;
- ✓ Понятие программы, подпрограммы, цикла, условия;
- ✓ Умение создавать программы и игры в Python;
- ✓ Умение аргументировать свою точку зрения и отстаивать ее;
- ✓ Умение создавать презентации.

Количество учебных часов/занятий, на которые рассчитан кейс (может варьироваться в зависимости от уровня подготовки, условий, и т.д.): **10 часов / 5 занятий.**

Ограничения:

Игра должна быть спроектирована в текстовом режиме с возможностью подгружать управлять длиной отрезка и выставлять ограничение на количество попыток.

Метод работы с кейсом: метод проектов.

Минимально необходимый уровень входных компетенций:

- ✓ Работа с компьютером на уровне начинающего пользователя
- ✓ Знание математики на уровне выпускника начальной школы

Необходимые расходные материалы и оборудование:

- ✓ Персональный компьютер с Windows 7 / 8 / 10 или Linux Debian 9.6

- ✓ Доступ в интернет
- ✓ Компилятор Python 3.5

Список рекомендуемых источников:

1. Марк Лутц – Изучаем Python и программируем на Python (2 тома).
2. Марк Саммерфилд - "Программирование на Python 3. Подробное руководство"

Педагогический сценарий (руководство для наставника)

Кейс представляет собой разработку игрового приложения для решения проблемной ситуации. Кейс включает в себя:

- ✓ Введение в проблему при помощи беседы с обучающимися (приведение жизненных примеров);
- ✓ Групповое обсуждение проблемы, поиск путей решения, введение в группу мысли о решении проблемы через разработку игрового приложения;
- ✓ Введение в программирование игровых приложений в Python;
- ✓ Составление технического задания на разработку игрового приложения;
- ✓ Сбор статистики и анализ результатов работы алгоритма;
- ✓ Подготовка к публичной презентации и защите проекта и защита проекта с демонстрацией игрового приложения;
- ✓ Подведение итогов, рефлексия.

Кейс игра «Виселица».

Описание проблемной ситуации.

«Виселица» – это игра для двоих, в которой один игрок загадывает слово и рисует на странице отдельные пустые клетки для каждой буквы. А второй игрок пытается угадать буквы, которые могут быть в данном слове, а затем и всё слово целиком.

Если второй игрок правильно угадывает букву, первый игрок вписывает её в соответствующую пустую клетку. А если ошибается, первый игрок рисует одну из частей тела повешенного человечка. Что бы победить, второй игрок должен угадать все буквы в слове до того, как повешенный человечек будет полностью нарисован.

Категория кейса: Вводный.

Место кейса в структуре модуля: Базовый, мотивационный кейс.

Ход работы:

Занятие 1		Занятие 2		Занятие 3	
Цель: Постановка проблемы, генерация путей решения		Цель: Изучение дизайна и механики игры		Цель: Визуализирование программы в виде блок-схемы	
Что делается: Представление поставленной проблемы группе детей. Анализ проблемы, генерация и обсуждение методов ее решения.	Компетенции: Умение искать информацию в различных источниках. Умение генерировать идеи предложенными методами.	Что делается: Знакомство с игровыми механиками игры, изучение ограничений и правил..	Компетенции: Работа с переменными величинами. Проверка наличия буквы в секретном слове. Проверка – не победил ли игрок. Обработка ошибочных предположений. Проверка – не проиграл ли игрок. Умение программировать в Python.	Что делается: Создание блок-схем. Ветвление в блок-схемах.	Компетенции: Умение генерировать идеи. Умение логически разбирать проект на блоки.
Занятие 4		Занятие 5		Занятие 6	

Цель: Тестирование и доработка.		Цель: Подготовка к публичному выступлению для защиты результатов		Цель: Демонстрация игры в группе и защита результатов	
Что делается: Тестирование созданной игры. Проверка на граничные условия. Выявление багов и их исправление.	Компетенции: Умение программировать на языке Python. Умение проводить тестирование созданных продуктов.	Что делается: Подготовка речи и презентации в TeX/LaTeX для публичной демонстрации результатов работы в кейсе.	Компетенции: Умение анализировать результаты работы. Грамотное представление результатов своей деятельности.	Что делается: Публичная презентация результатов работы. Ответы на вопросы жюри. Рефлексия.	Компетенции: Базовые навыки ораторства, публичных выступлений, аргументирование точки зрения.

Предполагаемые образовательные результаты учащихся (артефакты, решения), формируемые навыки (soft и hard skills):

- ✓ Развитие аналитического и алгоритмического мышления;
- ✓ Понятие алгоритма, последовательного выполнения действий;
- ✓ Понятие программы, подпрограммы, цикла, условия;
- ✓ Умение создавать программы и игры в Python;
- ✓ Умение аргументировать свою точку зрения и отстаивать ее;
- ✓ Умение создавать презентации.

Количество учебных часов/занятий, на которые рассчитан кейс (может варьироваться в зависимости от уровня подготовки, условий, и т.д.): **12 часов / 6 занятий.**

Ограничения:

Игра должна быть спроектирована в текстовом режиме с возможностью подгружать словари слов различной длины. Игра должна состоять минимум из трех уровней разной степени сложности.

Метод работы с кейсом: метод проектов.

Минимально необходимый уровень входных компетенций:

- ✓ Работа с компьютером на уровне начинающего пользователя
- ✓ Знание математики на уровне выпускника начальной школы

Необходимые расходные материалы и оборудование:

- ✓ Персональный компьютер с Windows 7 / 8 / 10 или Linux Debian 9.6

- ✓ Доступ в интернет
- ✓ Компилятор Python 3.5

Список рекомендуемых источников:

3. Марк Лутц – Изучаем Python и программируем на Python (2 тома).
4. Марк Саммерфилд - "Программирование на Python 3. Подробное руководство"

Педагогический сценарий (руководство для наставника)

Кейс представляет собой разработку игрового приложения для решения проблемной ситуации. Кейс включает в себя:

- ✓ Введение в проблему при помощи беседы с обучающимися (приведение жизненных примеров);
- ✓ Групповое обсуждение проблемы, поиск путей решения, введение в группу мысли о решении проблемы через разработку игрового приложения;
- ✓ Введение в программирование игровых приложений в Python;
- ✓ Составление технического задания на разработку игрового приложения;
- ✓ Создание уровней, программирование механики игры;
- ✓ Проведение тестирования разработанного игрового приложения и его доработка;
- ✓ Подготовка к публичной презентации и защите проекта и защита проекта с демонстрацией игрового приложения;
- ✓ Подведение итогов, рефлексия.

Кейс «Игра Крестики-Нолики»

Описание проблемной ситуации или феномена.

В игру играют два человека (или два компьютера). Один игрок рисует крестики, другой – нолики. Игроки по очереди размещают свои знаки. Если игрок разместил на игровом поле три своих знака подряд – в один ряд, в один столбец или по диагонали, - он побеждает. Если поле заполнено, но ни один игрок не выиграл, игра заканчивается ничьей.

Пользователь будет играть против искусственного интеллекта. Искусственный интеллект – это компьютерная программа, способная разумно отвечать на действия игрока.

Категория кейса: Вводный.

Место кейса в структуре модуля: Базовый, мотивационный кейс.

Ход работы (Возраст: IT-START 13-14 лет):

Занятие 1		Занятие 2		Занятие 3	
Цель: Постановка проблемы, генерация путей решения		Цель: Написание текстовой версии с искусственным интеллектом на языке Python		Цель: Обработка статистики, вывод и обработка результатов на языке Python.	
Что делается: Проводим анализ проблемной ситуации, генерация и обсуждение методов ее решения и возможности достижения конечного результата. Проводим разделение на группы.	Компетенции: Развитие критического мышления. Умение поиска и представления информации. Коммуникативность. Умение генерировать идеи, слушать и слышать собеседника, аргументированно отстаивать свою точку зрения. Развитие критического мышления, коммуникативных	Что делается: Разработка текстовой версии с искусственным интеллектом, игра человек с компьютером, компьютер с компьютером	Компетенции: Умения по разработке и написанию блоков игры: вывод игрового поля на экран, предоставлению игроку выбору кем ходить, размещение меток на игровом поле, дублирование данных игрового поля, вычисление по короткой схеме, создание искусственного	Что делается: Разработка и создание главного меню игры, реализация подсчета очков. Знакомство со структурой программы, основными типами данных.	Компетенции: Развитие алгоритмического мышления. Навыки работы с языком программирования Python.

	навыков, навыков командной работы.		интеллекта.		
Занятие 4		Занятие 5		Занятие 6	
Цель: Тестирование и доработка		Цель: Оформление презентационной игры. 3D-печать		Цель: Подготовка к публичному выступлению для защиты результатов	
Что делается: Тестирование созданной игры. Проверка на граничные условия. Выявление багов и их исправление. решения проблемы; - оформляют проектную идею и презентуют ее; - составляют сетевой график (план работы).	Компетенции: Умение программировать на языке Python. Умение проводить тестирование созданных продуктов.	Что делается: Проектирование рекламной 3D модели игры.	Компетенции: Умение конвертировать 3D модель в формат для принтера на печать. Умение настраивать нити и опции ПО 3D-принтера.	Что делается: Подготовка речи и презентации в TeX/LaTeX для публичной демонстрации результатов работы в кейсе.	Компетенции: Умение анализировать результаты работы. Грамотное представление результатов своей деятельности.
Занятие 7					
Цель: Демонстрация игры в группе и защита результатов					
Что делается: Публичная презентация результатов работы. Ответы на вопросы жюри. Рефлексия.	Компетенции: Базовые навыки ораторства, публичных выступлений, аргументирование точки зрения.				

Ход работы (Возраст: IT-TOTAL 15-17 лет):

Занятие 1		Занятие 2		Занятие 3	
Цель: Постановка проблемы, генерация путей решения		Цель: Написание текстовой версии с искусственным интеллектом на языке Python		Цель: Доработка искусственного интеллекта. Управление игровым циклом	
Что делается: Проводим анализ проблемной	Компетенции: Развитие критического мышления.	Что делается: Разработка текстовой версии с	Компетенции: Умения по разработке и написанию	Что делается: Выбор лучшего хода из списка. Проверка	Компетенции: Умения по разработке и написанию

ситуации, генерация и обсуждение методов ее решения и возможности достижения конечного результата. Проводим разделение на группы.	Умение поиска и представления информации. Коммуникативность. Умение генерировать идеи, слушать и слышать собеседника, аргументированно отстаивать свою точку зрения. Развитие критического мышления, коммуникативных навыков, навыков командной работы.	искусственным интеллектом, игра человек с компьютером, компьютер с компьютером	блоков игры: вывод игрового поля на экран, предоставлению игроку выбору кем ходить, размещение меток на игровом поле, дублирование данных игрового поля, вычисление по короткой схеме, создание искусственного интеллекта.	угловых, центральной и боковой клеток.	блоков игры, создание искусственного интеллекта.
Занятие 4		Занятие 5		Занятие 5	
Цель: Игровая аналитика ИИ.		Цель: Обработка статистики, вывод и обработка результатов на языке Python.		Цель: Тестирование и доработка	
Что делается: Оценка результативности и сделанного хода.	Компетенции: Умения по разработке и написанию блоков игры, создание искусственного интеллекта.	Что делается: Разработка и создание главного меню игры, реализация подсчета очков. Знакомство со структурой программы, основными типами данных.	Компетенции: Развитие алгоритмического мышления. Навыки работы с языком программирования Python.	Что делается: Тестирование созданной игры. Проверка на граничные условия. Выявление багов и их исправление. решения проблемы; - оформляют проектную идею и презентуют ее; - составляют сетевой график (план работы).	Компетенции: Умение программировать на языке Python. Умение проводить тестирование созданных продуктов.
Занятие 6		Занятие 7		Занятие 9	
Цель: Оформление презентационной игры. 3D-печать		Цель: Подготовка к публичному выступлению для защиты результатов		Цель: Демонстрация игры в группе и защита результатов	
Что делается: Проектирование	Компетенции: Умение	Что делается: Подготовка речи	Компетенции: Умение	Что делается: Публичная	Компетенции: Базовые

рекламной 3D модели игры.	конвертировать 3D модель в формат для принтера на печать. Умение настраивать нити и опции ПО 3D-принтера.	и презентации в TeX/LaTeX для публичной демонстрации результатов работы в кейсе.	анализировать результаты работы. Грамотное представление результатов своей деятельности.	презентация результатов работы. Ответы на вопросы жюри. Рефлексия.	навыки ораторства, публичных выступлений, аргументирование точки зрения.
---------------------------	---	--	--	--	--

Предполагаемые образовательные результаты учащихся (артефакты, решения), формируемые навыки (soft и hard skills).

Soft skills:

- ✓ умения слушать и слышать (коммуникация);
- ✓ умение генерировать идеи (креативность);
- ✓ умение работать в команде (коллаборация);
- ✓ умение грамотно формулировать свои мысли;
- ✓ развитие критического мышления;
- ✓ умение объективно оценивать результаты работы;
- ✓ развитие основ ораторского искусства и опыта публичных выступлений.

Hard skills:

- ✓ умение составлять алгоритм программы и формализовывать его в виде блок схемы;
- ✓ умение писать код программы на языке Python;

Количество учебных часов/занятий, на которые рассчитан кейс (может варьироваться в зависимости от уровня подготовки, условий, и т.д.): 14 часов / 7 занятий (группа IT-START 13-14 лет) и 18 часов/9 занятий (группа IT-TOTAL 15-17 лет).

Метод работы с кейсом: метод проектов.

Необходимые расходные материалы и оборудование:

- ✓ Персональный компьютер с Windows 7 / 8 / 10 или Linux Debian 9.6
- ✓ Доступ в интернет
- ✓ Компилятор Python 3.5

Список рекомендуемых источников:

1. Марк Лутц – Изучаем Python и программируем на Python (2 тома).
2. Марк Саммерфилд - "Программирование на Python 3. Подробное руководство"
3. Стивен С. Скиена – Алгоритмы. Руководство по разработке, 2011 г.
4. Генри С. Уоррен мл. – Алгоритмические трюки для программистов, 2014 г.
5. Джордж Хайнеман, Гари Поллис, Стэнли Селков – Алгоритмы. Справочник с примерами на C, C++, Java и Python, 2017 г.

Педагогический сценарий (руководство для наставника)

Кейс представляет собой разработку игрового приложения для решения проблемной ситуации. Кейс включает в себя:

- 1) Погружение учащихся в проблему кейса, поиск решения методом мозгового штурма, обсуждение и распределение учащихся по группам.
- 2) Знакомство учащихся с основными понятиями и принципами разработки игровых алгоритмов;
- 3) Рассказ об основах языка программирования C++.
- 4) Организация работы групп: распределение ролей, формирование проектной идеи и программы работы каждой группы.
- 5) В ходе конструирования решения наставник находится в роли наблюдателя, а в случае сложностей – подтолкнуть к верному ответу.
- 6) Помощь в проведении презентации получившейся системы «Умная лампочка» и подведении итогов работы.
- 7) Рефлексия работы над кейсом.

Рефлексия (заготовленные вопросы):

Понятна ли была структура составленной программы? Какие сложности возникли во время написания программы? Как вы с ними справились? Какие еще можно усовершенствовать алгоритм работы компьютера? Что будет если заставить играть компьютер с компьютером?

Кейс «Игра Реверси»

Описание проблемной ситуации.

В игре «Реверси» используется поле размером 8x8 клеток и фишки – черные с одной стороны и белые с другой.

Два игрока по-очереди выставляют на поле фишки выбранного ими цвета – черные или белые. Когда игрок помещает фишку на поле, все фишки противника, которые находятся между новой фишкой и остальными фишками игрока, переворачиваются. Цель игры состоит в том, чтобы фишек вашего цвета было больше, чем фишек противника.

Искусственный интеллект здесь будет более продвинутой, чем у игры «Крестики-Нолики». Он, скорее всего, всегда будет побеждать игрока.

Категория кейса: Углубленный.

Ход занятия:

Занятие 1		Занятие 2		Занятие 3	
Цель: Постановка проблемы, генерация путей решения		Цель: Освоение языка Python		Цель: Освоение языка Python	
Что делается: Представление поставленной проблемы группе детей. Анализ проблемы, генерация и обсуждение методов ее решения.	Компетенции: Умение искать информацию в различных источниках. Умение генерировать идеи предложенными методами.	Что делается: Изучаемые понятия: циклы, функции. Ввод - вывод с текстовыми файлами.	Компетенции: Работа с циклами в Python. Работа с функциями в Python. Умение программировать на языке Python.	Что делается: Подключение модулей в языке python. Установка дополнительных модулей в систему.	Компетенции: Работа с внешними библиотеками в Python. Умение программировать на языке Python.
Занятие 4		Занятие 5		Занятие 6	
Цель: Освоение структуры проекта		Цель: Освоение структуры проекта		Цель: Изучение игровой механики игры	
Что делается: Структура данных игрового поля. Проверка допустимости хода. Проверка допустимости координат.	Компетенции: Умение программировать алгоритмы на языке Python.	Что делается: Получение игрового счета. Определение первого игрока.	Компетенции: Умение программировать алгоритмы на языке Python.	Что делается: Помещение фишки на поле. Создание копии структуры данных игрового поля.	Компетенции: Умение программировать алгоритмы на языке Python.
Занятие 7		Занятие 8		Занятие 9	

Цель: Изучение игровой механики игры		Цель: Изучение игровой механики игры		Цель: Изучение игровой механики игры	
Что делается: Получение хода игрока. Вывод игрового счета на экран. Игровой цикл. Реализация.	Компетенции: Умение программировать алгоритмы на языке Python.	Что делается: Компьютер против компьютера. Удаление приглашений для игрока и добавление игрока-компьютера.	Компетенции: Умение программировать алгоритмы на языке Python.	Что делается: Отслеживание результатов партий. Оценка искусственных интеллектов в процентном соотношении.	Компетенции: Умение программировать алгоритмы на языке Python.
Занятие 10		Занятие 11		Занятие 12	
Цель: Изучение искусственного интеллекта игры		Цель: Изучение искусственного интеллекта игры		Цель: Изучение искусственного интеллекта игры	
Что делается: Сравнение различных алгоритмов ИИ. ИИ лучшего углового хода. ИИ худшего хода. ИИ случайного хода.	Компетенции: Умение программировать алгоритмы на языке Python.	Что делается: Сравнение различных алгоритмов ИИ. ИИ худшего хода против ИИ лучшего хода.	Компетенции: Умение программировать алгоритмы на языке Python.	Что делается: Сравнение различных алгоритмов ИИ. ИИ случайного хода против ИИ лучшего игрового хода.	Компетенции: Умение программировать алгоритмы на языке Python.
Занятие 13		Занятие 14		Занятие 15	
Цель: Изучение искусственного интеллекта игры		Цель: Анализ статистики сыгранных партий		Цель: Оформление презентационной игры. 3D-печать	
Что делается: Сравнение различных алгоритмов ИИ. ИИ лучшего углового-граничного хода против ИИ лучшего игрового хода.	Компетенции: Умение программировать алгоритмы на языке Python.	Что делается: Сохранение и анализ результатов игры. Оптимизация искусственного интеллекта.	Компетенции: Работа с внешними библиотеками в Python. Умение программировать алгоритмы на языке Python.	Что делается: Проектирование рекламной 3D модели игры.	Компетенции: Умение конвертировать 3D модель в формат для принтера на печать. Умение настраивать нити и опции ПО 3D-принтера.
Занятие 16		Занятие 17		Занятие 18	
Цель: Тестирование и доработка.		Цель: Подготовка к публичному выступлению для защиты результатов		Цель: Демонстрация программы в группе и защита результатов	
Что делается: Тестирование	Компетенции: Умение	Что делается: Подготовка	Компетенции: Умение	Что делается: Публичная	Компетенции: Базовые навыки

созданной программы. Проверка на граничные условия. Выявление багов и их исправление.	программировать на языке Python. Умение проводить тестирование созданных продуктов.	речи и презентации в TeX/LaTeX для публичной демонстрации результатов работы в кейсе.	анализировать результаты работы. Грамотное представление результатов своей деятельности.	презентация результатов работы. Ответы на вопросы жюри. Рефлексия.	ораторства, публичных выступлений, аргументирование точки зрения.
---	---	---	--	--	---

Предполагаемые образовательные результаты учащихся (артефакты, решения), формируемые навыки (soft и hard skills):

- ✓ Понятие алгоритма, последовательного выполнения действий;
- ✓ Понятие программы, подпрограммы, цикла, условия;
- ✓ Понятие одномерного и многомерного массива;
- ✓ Понятие кодировки, файлового ввода-вывода;
- ✓ Умение создавать и выполнять программы на языке Python в консоли и в специализированных редакторах кода;
- ✓ Умение создавать искусственный интеллект;
- ✓ Умение управлять искусственным интеллектом;
- ✓ Умение аргументировать свою точку зрения и отстаивать ее.

*Количество учебных часов/занятий, на которые рассчитан кейс (может варьироваться в зависимости от уровня подготовки, условий, и т.д.): **36 часов / 18 занятий***

Ограничения: Программа должна быть написана таким образом, чтобы посторонний человек не мог увидеть ходы компьютера.

Метод работы с кейсом: Метод проектов

Минимально необходимый уровень входных компетенций:

- ✓ Знание операционных систем Unix/Linux
- ✓ Знание продвинутых алгоритмов языка Python
- ✓ Знание математики (алгебры и геометрии) на уровне 8 класса школы.

Необходимые расходные материалы и оборудование:

- ✓ Персональный компьютер с Windows 7 / 8 / 10 или Linux Debian 9.6
- ✓ Доступ в интернет

- ✓ Установленный Python 3.5

Список рекомендуемых источников:

1. Python для детей. Самоучитель по программированию / Джейсон Бриггс; пер. с англ. Станислава Ломакина; [науч. ред. Д. Абрамова]. — М.: Манн, Иванов и Фербер, 2017. — 320 с.
2. Little Python - видеоуроки по программированию на языке Python — https://www.youtube.com/watch?v=AOy65ssV-3I&list=PL_sNWIjWO0MaAmycAs6eTromID8uzqG2S
3. Курс «Д. П. Кириенко. Программирование на языке Python (школа 179 г. Москвы)» — <https://informatics.msk.ru/course/view.php?id=156>
4. Программирование аркадных игр и обучение информатике — <http://programarcadegames.com/index.php?lang=ru>

Педагогический сценарий (руководство для наставника)

Кейс представляет собой разработку игрового приложения для решения проблемной ситуации. Кейс включает в себя:

- ✓ Введение в проблему при помощи беседы с обучающимися (приведение жизненных примеров);
- ✓ Групповое обсуждение проблемы, поиск путей решения, введение в группу мысли о решении проблемы через разработку игрового приложения;
- ✓ Изучение основ языка Python, выполнение команд в интерпретаторе;
- ✓ Изучение понятий «переменные», «строки», «условия», «циклы», «массивы», «функции», «модули»;
- ✓ Проведение тестирования разработанного игрового приложения и его доработка;
- ✓ Подготовка к публичной презентации и защите проекта и защита проекта с демонстрацией игрового приложения;
- ✓ Подведение итогов, рефлексия.

Кейс «Игра Ловкач»

Описание проблемной ситуации.

В игре «Ловкач» игрок управляет спрайтом - «своим» персонажем, который должен уклониться от злодеев (таких же спрайтов), падающих с верхней части экрана. Чем дольше игрок сможет уклоняться, тем больше очков он наберет. Предлагается написать приложение, которое будет имитировать работу нескольких искусственных интеллектов для обхода препятствий, для выяснения какая из программ работает оптимальнее.

Категория кейса: Углубленный.

Место кейса в структуре модуля: Кейс на углубленное изучение языка Python и получение знаний по теории игр.

Занятие 1		Занятие 2		Занятие 3	
Цель: Постановка проблемы, генерация путей решения		Цель: Создание прототипа игры для запуска искусственных интеллектов.		Цель: Создание прототипа игры для запуска искусственных интеллектов.	
Что делается: Представление поставленной проблемы группе детей. Анализ проблемы, генерация и обсуждение методов ее решения.	Компетенции: Умение искать информацию в различных источниках. Умение генерировать идеи предложенными методами.	Что делается: Создание главного окна программы, создание архитектуры программы, создание юнитов.	Компетенции: Работа с внешними библиотеками в Python. Работа с графикой в Python. Умение программировать на языке Python	Что делается: Установка pygame. Инициализация pygame. Настройка окна pygame. Кортежи. Объекты поверхности.	Компетенции: Работа с внешними библиотеками и в Python. Работа с графикой в Python. Умение программировать на языке Python
Занятие 4		Занятие 5		Занятие 6	
Цель: Создание прототипа игры для запуска искусственных интеллектов.		Цель: Создание прототипа игры для запуска искусственных интеллектов.		Цель: Создание прототипа игры для запуска искусственных интеллектов.	
Что делается: Работа с цветом. Вывод текста в окне pygame. Использование шрифтов для оформления	Компетенции: Работа с внешними библиотеками в Python. Работа с графикой в Python. Умение	Что делается: Функции рисования pygame. Рисование многоугольника. Рисование линии. Рисование круга. Рисование эллипса.	Компетенции: Работа с внешними библиотеками в Python. Работа с графикой в Python. Умение	Что делается: Анимированная графика. Перемещение и контроль отскока блоков. Константы для направлений.	Компетенции: Работа с внешними библиотеками и в Python. Работа с

текста. Рендеринг объекта. Настройка местоположения текста с помощью атрибутов.	программировать на языке Python	Рисование прямоугольника. Окрашивание пикселей.	программировать на языке Python	Константы для цвета. Создание структуры данных блока. Перемещение каждого блока. Управление отскакиванием блока. Отображение блоков в новых положениях.	графикой в Python. Умение программировать на языке Python
Занятие 7		Занятие 8		Занятие 9	
Цель: Создание прототипа игры для запуска искусственных интеллектов.		Цель: Создание прототипа игры для запуска искусственных интеллектов.		Цель: Создание прототипа игры для запуска искусственных интеллектов.	
Что делается: Анимированная графика. Перемещение и контроль отскока блоков. Константы для направлений. Константы для цвета. Создание структуры данных блока. Перемещение каждого блока. Управление отскакиванием блока. Отображение блоков в новых положениях.	Компетенции: Работа с внешними библиотеками в Python. Работа с графикой в Python. Умение программировать на языке Python	Что делается: Добавление изображений с помощью спрайтов. Графические и звуковые файлы. Установка музыки и звуков. Добавление аудиофайлов. Включение/отключение музыки.	Компетенции: Работа с внешними библиотеками в Python. Работа с графикой в Python. Умение программировать на языке Python	Что делается: Определение функций. Завершение игры и добавление паузы. Отслеживание столкновений. Отображение текста в окнах. Игровой цикл. Обработка событий клавиатуры. Обработка событий мыши.	Компетенции: Работа с внешними библиотеками в Python. Работа с графикой в Python. Умение программировать на языке Python
Занятие 10		Занятие 11		Занятие 12	
Цель: Создание прототипа игры для запуска искусственных интеллектов.		Цель: Создание прототипа игры для запуска искусственных интеллектов.		Цель: Создание искусственного интеллекта для игры	
Что делается: Инициализация и настройка окна. Установка шрифтов, изображений и	Компетенции: Работа с внешними библиотеками в Python. Работа с графикой в Python. Умение	Что делается: Создание карты, создание механики юнитов.	Компетенции: Работа с внешними библиотеками в Python. Работа с графикой в Python. Умение	Что делается: Создание базового искусственного интеллекта, способного управлять юнитами.	Компетенции: Работа с внешними библиотеками в Python. Умение

звук. Отображение начального экрана. Реализация чит- кодов. Экран окончания игры. Измерение игры.	программировать на языке Python		программировать на языке Python		программиро вать на языке Python
Занятие 13		Занятие 14		Занятие 15	
Цель: Создание искусственного интеллекта для игры		Цель: Проведение соревнований искусственных интеллектов		Цель: Оформление презентационной игры. 3D-печать	
Что делается: Расширение возможностей искусственного интеллекта, способного стремиться к победе.	Компетенции: Работа с внешними библиотеками в Python. Умение программировать на языке Python.	Что делается: Копирование искусственных интеллектов на главный компьютер, запуск соревнований, составление турнирной таблицы.	Компетенции: Работа с внешними библиотеками в Python. Умение программировать на языке Python.	Что делается: Проектирование рекламной 3D модели игры.	Компетенции: Умение конвертиров ать 3D модель в формат для принтера на печать. Умение настраивать нити и опции ПО 3D- принтера.
Занятие 16		Занятие 17		Занятие 18	
Цель: Тестирование и доработка. Повторное проведение соревнований.		Цель: Подготовка к публичному выступлению для защиты результатов		Цель: Демонстрация программы в группе и защита результатов	
Что делается: Тестирование созданной программы. Проверка на граничные условия. Выявление багов и их исправление.	Компетенции: Умение программировать на языке Python. Умение проводить тестирование созданных продуктов.	Что делается: Подготовка речи и презентации в TeX/LaTeX для публичной демонстрации результатов работы в кейсе.	Компетенции: Умение анализировать результаты работы. Грамотное представление результатов своей деятельности.	Что делается: Публичная презентация результатов работы. Ответы на вопросы жюри. Рефлексия.	Компетенции: Базовые навыки ораторства, публичных выступлений , аргументиро вание точки зрения.

Предполагаемые образовательные результаты учащихся (артефакты, решения), формируемые навыки (soft и hard skills):

- ✓ Понятие алгоритма, последовательного выполнения действий;
- ✓ Понятие программы, подпрограммы, цикла, условия;

- ✓ Понятие одномерного и многомерного массива;
- ✓ Понятие кодировки, файлового ввода-вывода;
- ✓ Умение создавать и выполнять программы на языке Python в консоли и в специализированных редакторах кода;
- ✓ Умение аргументировать свою точку зрения и отстаивать ее;
- ✓ Умение создавать презентации.

Количество учебных часов/занятий, на которые рассчитан кейс (может варьироваться в зависимости от уровня подготовки, условий, и т.д.): **36 часов / 18 занятий**

Метод работы с кейсом: Метод проектов.

Минимально необходимый уровень входных компетенций:

- ✓ Знание операционных систем Unix/Linux
- ✓ Знание продвинутых алгоритмов языка Python
- ✓ Знание математики (алгебры и геометрии) на уровне 8 класса школы.

Необходимые расходные материалы и оборудование:

- ✓ Персональный компьютер с Windows 7 / 8 / 10 или Linux Debian 9.6
- ✓ Доступ в интернет
- ✓ Установленный Python 3.5

Список рекомендуемых источников:

1. Python для детей. Самоучитель по программированию / Джейсон Бриггс; пер. с англ. Станислава Ломакина; [науч. ред. Д. Абрамова]. — М.: Манн, Иванов и Фербер, 2017. — 320 с.
2. Little Python - видеоуроки по программированию на языке Python — https://www.youtube.com/watch?v=AOy65ssV-3I&list=PL_sNWIjWO0MaAmycAs6eTromID8uzqG2S
3. Программирование аркадных игр и обучение информатике — <http://programarcadegames.com/index.php?lang=ru>

Педагогический сценарий (руководство для наставника)

Кейс представляет собой разработку игрового приложения для решения проблемной ситуации. Кейс включает в себя:

- ✓ Введение в проблему при помощи беседы с обучающимися (приведение жизненных примеров);
- ✓ Групповое обсуждение проблемы, поиск путей решения, введение в группу мысли о решении проблемы через разработку игрового приложения;
- ✓ Создание программы;
- ✓ Проведение тестирования разработанного игрового приложения и его доработка;
- ✓ Подготовка к публичной презентации и защите проекта и защита проекта с демонстрацией игрового приложения;
- ✓ Подведение итогов, рефлексия.

ТЕХНОЛОГИЧЕСКАЯ КАРТА ОПРЕДЕЛЕНИЯ УРОВНЯ ОСВОЕНИЯ ОБУЧАЮЩИМИСЯ ДОПОЛНИТЕЛЬНОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Мониторинг личностного развития ребенка

Показатели (оцениваемые параметры)	Критерии	Степень выраженности оцениваемого качества	Уровень
1.Организационно-волевое качество			
Терпение	Способность переносить нагрузки в течение определенного времени, преодолевать трудности	-терпения хватает меньше чем на ½ занятия	низкий
		-терпения хватает больше чем на ½ занятия	средний
		- терпения хватает на все занятие	высокий
Воля	Способность активно побуждать себя к практическим действиям	- волевые усилия ребенка побуждаются извне	низкий
		- иногда сами ребенком	средний
		- всегда самим ребенком	высокий
2. Ориентационные качества			
Интерес к занятиям в детском объединении	Осознанное участие ребенка в освоении образовательной программы	- интерес к занятиям продиктован ребенку извне	низкий
		- интерес периодически поддерживается самим ребенком	средний
		- интерес постоянно поддерживается ребенком самостоятельно	высокий
3 Поведенческие качества			
Конфликтность (отношение ребенка к столкновению интересов, спору)	Способность занять определенную позицию в конфликтной ситуации	- периодически провоцирует конфликты	низкий
		- сам в конфликтах не участвует, старается их избегать	средний
		- пытается самостоятельно уладить возникающие конфликты	высокий
Тип сотрудничества (отношение ребенка к общим делам)	Умение воспринимать общие дела, как свои собственные	- избегает участия в общих делах	низкий
		- участвует при побуждении извне	средний
		- инициативен в общих делах	высокий

**Письменный тест по теме «Операционная система Linux»
(к промежуточной аттестации)**

1. Какой каталог Linux содержит наиболее важные программы и команды?

- а) /bin*
- б) /dev*
- в) /lib*
- г) /mnt*

2. Какая из перечисленных команд используется для поиска в файле заданной строки и отображения её на экране?

- а) cat*
- б) less*
- в) grep*
- г) cp*

3. Какие действия выполняет следующая команда?

`mv file1 file2`

- а) копирует *file1* и создаёт *file2**
- б) переименовывает *file1* в *file2**
- в) создаёт *file2* и удаляет *file1**
- г) сравнивает *file1* и *file2**

4. Какая команда текстового редактора `vi` позволяет удалить символ в позиции курсора и сдвинуть следующие за ним символы влево?

- а) x*
- б) X*
- в) D*
- г) dd*

5. Как выйти из текстового редактора vi без сохранения текста?

а) q:

б) :q!

в) :wq

г) :w

6. Какие права будут установлены для файла readme.txt в результате выполнения следующей команды?

\$ chmod -aw readme.txt

а) -rwxrwxrwx

б) -rwxrwxr-x

в) -r--r--r--

г) -rw-rw-r--

7. Что обозначают права доступа к файлу, имеющие значение 664?

а) Пользователь и группа имеют право на чтение и запись, а остальные имеют право только на запись.

б) Пользователь и группа имеют право на чтение и выполнение, а остальные имеют право только на чтение.

в) Пользователь и группа имеют права на чтение, выполнение и запись, а все остальные – только на чтение.

г) Пользователь и группа имеют право на чтение и запись, а все остальные – только на чтение.

8. Выберите верное утверждение?

а) Для добавления пользователя в систему достаточно команды useradd.

б) Чтобы добавить пользователя в систему используются две команды useradd и passwd.

в) Чтобы добавить в систему нового пользователя не обязательно иметь права пользователя root.

г) Для удаления пользователя используется команда `deluser`.

9. Какую из перечисленных команд необходимо использовать, чтобы перезагрузить компьютер?

а) `halt`

б) `shutdown`

в) `reboot`

г) `shutdown -h 0`

10. Как обозначается домашний каталог?

а) `/`

б) `~`

в) `..`

г) `$`

Список письменных вопросов по пройденным темам (к промежуточной аттестации)

1. Какие языки программирования существуют, в чем их отличие друг от друга?
2. Переменные в языке Python: создание переменной в программе, изменение переменной, вывод на экран, ввод с клавиатуры.
3. Синтаксис условий на языке Python: условия `if`, условия `if-else`, условия `if-elif-else`.
4. Синтаксис циклов на языке Python: цикл `while`, цикл `for`, условие выполнения цикла.
5. Синтаксис массивов на языке Python: создание массива, добавление элементов в массив.
6. Работа с файлами: открытие файла для чтения и для записи, запись в файл, чтение из файла.

7. Написать программу на языке Python, которая будет просить ввести число с клавиатуры, и выводить на экран, является введенное число четным или нет.
8. Написать программу на языке Python, которая будет находить количество элементов массива, самый большой элемент в массиве, и самый маленький элемент в массиве.
9. Написать программу на языке Python, которая будет выводить на экран массив из 10000 случайных чисел в промежутке от 1000 до 100000.
10. Написать программу на языке Python, которая будет подсчитывать количество слов в абзаце текста.

**Список письменных вопросов по пройденным темам
(к аттестации по итогам реализации программы)**

11. Синтаксис языка Python. Переменные, циклы, условия.
12. Работа с файлами: открытие файла для чтения и для записи, запись в файл, чтение из файла.
13. Работа с изображениями в языке Python. Открытие изображений, вывод на экран, сохранение, перевод в оттенки серого.
14. Одномерные и двумерные массивы в Python. Задачи, в которых используются одномерные и двумерные массивы.
15. Написать программу, которая просит пользователя ввести с клавиатуры размер массива, создает массив этого размера из случайных чисел, находит сумму всех элементов массива, самый большой элемент в массиве, и самый маленький элемент в массиве.
16. Написать программу на языке Python, которая будет подсчитывать количество слов в абзаце текста.
17. Написать программу, которая определяет, является ли слово или предложение палиндромом.
18. Написать программу, которая оставляет в тексте только согласные буквы.

19. Написать программу, которая считывает из текстового файла двумерный массив, производит какие-либо вычисления над элементами и сохраняет обратно в файл.
20. Написать программу, которая открывает изображение, переводит в оттенки серого, поворачивает на 45 градусов против часовой стрелки, и сохраняет обратно в массив.

Параметры и критерии защиты итогового проекта

Параметры	Низкий	Средний	Высокий
1. Оригинальность темы и идеи проекта	Тема неактуальна и не соответствует возрастным особенностям и полученным программным знаниям. Нет плана работы над проектом, программа примитивна и выполнена небрежно	Тема проекта недостаточно актуальна и значима, но творчески интересна. Знает порядок создания алгоритма программы, имеет план работы над проектом	Выбор актуальной темы проекта, его логическое обоснование, наличие плана работы по выполнению проекта.
2. Техническое решение	Слабое владение основными навыками программирования	Достаточно владеет навыками программирования	Техническое решение воплощено в полной мере в действующей модели
3. Защита проекта	Неясные умозаключения, неумение рассказать об результатах разработки	Рассказывает про созданную программу, но не умеет отвечать на вопросы	В четкой логической последовательности излагает мысли, анализирует информацию и отстаивает свою точку зрения
4. Проявляемый интерес к занятиям, творческая активность	Минимальный интерес. Безынициативен, работает сам по себе, замечания принимает враждебно, всегда предъявляет претензии, отсутствует коммуникативный опыт защиты проекта	Интерес стабильный. Недостаточно инициативен в совместном творчестве, присутствует дружелюбность в общении с товарищами, присутствует ответственность за общее дело, деловитость, не достаточно полно согласует свои действия с действиями партнера.	Бесконфликтно и инициативно работает в команде, эффективно распределяют обязанности внутри команды. Участие в соревнованиях и фестивалях.