


Министерство образования и науки Мурманской области  
Государственное автономное учреждение дополнительного образования  
Мурманской области «Мурманский областной центр  
дополнительного образования «Лапландия»

ПРИНЯТА  
методическим советом  
Протокол  
от 16.05.2021 № 42

Председатель  А.Ю. Решетова

УТВЕРЖДЕНА  
приказом ГАУДОМО  
«МОЦДО «Лапландия»  
от 16.05.2021 № 644

Директор  С.В. Кулаков



## IT-КВАНТУМ

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ  
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА  
ТЕХНИЧЕСКОЙ НАПРАВЛЕННОСТИ  
«Введение в разработку программ и компьютерных игр.  
Линия 2»

Возраст учащихся: **14-16 лет**  
Срок реализации программы: **1 год**

Авторы-составители:  
**Патрикеева Ольга Николаевна,**  
**Шуньгина Ирина Владимировна,**  
педагоги дополнительного образования

Мурманск  
2021

## **I. Пояснительная записка**

### Область применения программы

Программа разработана в рамках 2 линии общего цикла «Введение в разработку программ и компьютерных игр» дополнительных общеобразовательных программ технической направленности направления «IT-Квантум» детского технопарка «Кванториум» и направлена на удовлетворение образовательных потребностей учащихся в области разработки программных продуктов для современных операционных систем, для учащихся в возрасте от 14 до 16 лет.

### Нормативно-правовая база разработки и реализации программы

Программа разработана в соответствии с нормативно-правовыми актами:

- Федеральным законом от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;
- приказом Министерства образования и науки РФ от 09.11.2018 № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- письмом Министерства образования и науки РФ от 25.07.2016 № 09-1790 «Рекомендации по совершенствованию дополнительных образовательных программ, созданию детских технопарков, центров молодежного инновационного творчества и внедрению иных форм подготовки детей и молодежи по программам инженерной направленности»;
- постановлением Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении Санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи».
- с постановлением Главного государственного санитарного врача РФ от 28.01.2021 №2 «Об утверждении санитарных правил и норм СанПиН 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания».

### Актуальность, педагогическая целесообразность реализации программы

Актуальность программы «Введение в разработку программ и компьютерных игр» Линия 2 вызвана потребностью современного информационного общества в высокообразованных, адаптированных к изменениям специалистах в IT-сфере. Традиционное изучение в общеобразовательных учреждениях языка программирования «Pascal» не может в полной мере удовлетворить потребности обучающегося информационно-технологической и физико-математической направленностей в изучении новых и популярных языков программирования. Изучение языка программирования «Python» поможет ребенку получить более целостное

представление о профессии программиста, разработчика, инженера. Программа предполагает участие обучающихся в интеллектуальных соревнованиях по программированию различных уровней, создание образовательных практических или научно-исследовательских проектов, что даст возможность детям полностью реализовать свой интеллектуальный и творческий потенциал. В ходе практических занятий по программе модуля «Хайтек» обучающиеся знакомятся с различными видами высокотехнологичного оборудования, изучают принципы его функционирования и возможности использования при решении конкретных прикладных задач, приобретают практические навыки работы на лазерном, фрезерном станках, 3D-принтерах. В ходе работы над кейсами учащиеся знакомятся с понятием изобретательской задачи, получают представление о методах их решения, в частности, о методе поиска инженерного решения, приобретают начальные знания о технологиях трехмерного моделирования, изучают принципы лазерных, аддитивных технологий производства.

Педагогическая целесообразность программы «Введение в разработку программ и компьютерных игр» Линия 2 заключается в привлечении учащихся к занятиям техническим творчеством, что способствует развитию логического мышления, творческих способностей и навыков решения задач программирования. Программирование мотивирует к занятиям в различных научных областях (физики, информатики, алгебры, геометрии и др.), развивает воображение и способствует ранней профориентации подростков. Для достижения поставленных задач занятия проводятся в формате «от простого к сложному». Учащиеся получают знания по основам алгоритмизации и программирования и на их основе, углубляя их, учатся составлять простые и сложные программы.

#### Цель программы

Освоение hard и soft компетенций учащимися в области программирования на языке высокого уровня на основе применения кейс-технологий.

#### Задачи программы (обучающие, развивающие, воспитательные)

##### Обучающие:

- получить представление о процессе разработки приложений для современных операционных систем;
- сформировать представление о средствах разработки;
- познакомить с общими принципами разработки электронных устройств;
- познакомить с инструментальными средствами программирования электронных устройств и приложений;
- научить создавать простейшие электронные устройства и приложения;
- привить навыки проектной деятельности.

##### Развивающие:

- формирование корректного лексического поля, отражающего направленность программы;
- способствовать развитию памяти, внимания, технического мышления, изобретательности;
- способствовать развитию алгоритмического мышления;

- способствовать формированию интереса к техническим знаниям;
- способствовать формированию умения практического применения полученных знаний;
- сформировать умение формулировать, аргументировать и отстаивать своё мнение ;
- сформировать умение выступать публично с докладами, презентациями и т. п.

Воспитательные:

- воспитание аккуратности и дисциплинированности при выполнении работы;
- способствовать формированию положительной мотивации к трудовой деятельности;
- способствовать формированию опыта совместного и индивидуального творчества при выполнении командных заданий;
- воспитывать трудолюбие, уважение к труду;
- формировать чувство коллективизма и взаимопомощи;
- воспитывать чувство патриотизма, гражданственности, гордости за достижения отечественной науки и техники.

Адресат программы

Учащиеся возраста 14-16 лет имеющие представление об основах программирования в визуальных средах, интересующиеся разработкой программных продуктов и основами электроники.  
Форма реализации программы

Очная

Срок освоения программы

1 учебный год, 162 часа.

Форма организации занятий

Групповая

Режим занятий

162 часа: 2 раза в неделю по 2 академических часа – IT-квантум

18 часов: 1 раз в неделю по 1 академическому часу – Хайтек цех

Количество учащихся в группе: 6-9 человек.

Виды учебных занятий и работ

Лекции, практические занятия, круглые столы, ролевые игры, и пр.

Ожидаемые результаты обучения

**Личностные результаты:**

- критическое отношение к информации и избирательность её 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;
- развитие самостоятельности суждений, независимости и нестандартности мышления;
- освоение социальных норм, правил поведения, ролей и форм социальной жизни в группах и сообществах;
- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками.

**Метапредметные результаты:**

*Регулятивные универсальные учебные действия:*

- умение принимать и сохранять учебную задачу;
- умение планировать последовательность шагов алгоритма для достижения цели;
- умение ставить цель (создание творческой работы), планировать достижение этой цели;
- умение осуществлять итоговый и пошаговый контроль по результату;
- способность адекватно воспринимать оценку педагога и сверстников;
- умение различать способ и результат действия;
- умение вносить коррективы в действия в случае расхождения результата решения задачи на основе ее оценки и учета характера сделанных ошибок;
- умение в сотрудничестве ставить новые учебные задачи;
- способность проявлять познавательную инициативу в учебном сотрудничестве;
- умение осваивать способы решения проблем творческого характера в жизненных ситуациях;
- умение оценивать получающийся творческий продукт и соотносить его с изначальным замыслом, выполнять по необходимости коррекции либо продукта, либо замысла.

*Познавательные универсальные учебные действия:*

- умение осуществлять поиск информации в индивидуальных информационных архивах учащегося, информационной среде образовательного учреждения, в федеральных хранилищах информационных образовательных ресурсов;
- умение использовать средства информационных и коммуникационных технологий для решения коммуникативных, познавательных и творческих задач;
- умение ориентироваться в разнообразии способов решения задач;
- умение осуществлять анализ объектов с выделением существенных и несущественных признаков;
- умение проводить сравнение, классификацию по заданным критериям;
- умение строить логические рассуждения в форме связи простых суждений об объекте;
- умение устанавливать аналогии, причинно-следственные связи;
- умение моделировать, преобразовывать объект из чувственной формы в модель, где выделены существенные характеристики объекта (пространственно-графическая или знаково-символическая);
- умение синтезировать, составлять целое из частей, в том числе самостоятельное достраивание с восполнением недостающих компонентов.

*Коммуникативные универсальные учебные действия:*

- умение аргументировать свою точку зрения на выбор оснований и критериев при выделении признаков, сравнении и классификации объектов;
- умение выслушивать собеседника и вести диалог;
- способность признавать возможность существования различных точек зрения и права каждого иметь свою;
- умение планировать учебное сотрудничество с педагогом и сверстниками: определять цели, функций участников, способов взаимодействия;
- умение осуществлять постановку вопросов: инициативное сотрудничество в поиске и сборе информации;
- умение разрешать конфликты: выявление, идентификация проблемы, поиск и оценка альтернативных способов разрешения конфликта, принятие решения и его реализация;
- умение с достаточной полнотой и точностью выражать свои мысли в соответствии с задачами и условиями коммуникации;
- владение монологической и диалогической формами речи.

**Предметные результаты:**

*Учащиеся будут иметь представление:*

- о механизмах разработки и проектирования программных продуктов;

- о проектировании и реализации приложений для современных операционных систем;
- о базовых электронных компонентах;
- о базовых алгоритмах программирования аппаратных устройств ввода\вывода;
- о назначения и возможностей современных систем автоматизированного проектирования (САПР);
- о базовых принципов построения изображений в векторной двумерной и трехмерной графике;
- о базовых принципов создания продукта с использованием высокотехнологичного оборудования.

*Учащиеся будут знать:*

- основные алгоритмические конструкции;
- основные инструменты разработки программных продуктов;
- принципы реализации модульной конструкции устройства;
- принципы разработки аппаратных и программных интерфейсов;
- виды различного высокотехнологичного оборудование и области его применения;
- потенциальные риски при работе с высокотехнологичным оборудованием и умение соблюдать технику безопасности.

*Учащиеся будут уметь:*

- разрабатывать простейшие программные продукты;
- анализировать современные электронные устройства;
- разрабатывать электронные компьютерные игровые системы;
- представлять свой проект или решение кейса.

Формы итоговой аттестации

Выставки, фестивали, конференции, защита проекта и другие.

## II. Учебный план

№ п/п	Название раздела, темы	Количество часов			Формы аттестации/ контроля
		Всего	Теория	Практика	

1	Введение в образовательный модуль, техника безопасности. Первичный инструктаж.	2	1	1	Беседа Тестирование
2	Кейс 1: «Вычисление периметра и площади заданного объекта».				Демонстрация решений кейса
3	Кейс 2: «Обмен значений переменных».				Демонстрация решений кейса
4	Кейс 3: «Определение существования треугольника по трем сторонам».				Демонстрация решений кейса
5	Кейс 4: «Принадлежит ли точка кругу?».				Демонстрация решений кейса
6	Кейс 5: «Простейший калькулятор».				Демонстрация решений кейса Тестирование
7	Кейс 6: «Алгоритм Евклида». Повторный инструктаж.				Демонстрация решений кейса
8	Кейс 7: «Проверка уникальности элементов списка».				Демонстрация решений кейса
9	Кейс 8: «Решето Эратосфена».				Демонстрация решений кейса
10	Кейс 9: «Выбор числа из строки».				Демонстрация решений кейса
11	Кейс 10: «Подсчет количества одинаковых элементов списка».				Демонстрация решений кейса Тестирование
12	Заключительное занятие				Беседа
13	Мероприятия, направленные на развитие общекультурных компетенций.				Участие в мероприятиях Демонстрация решений кейса
	<b>Итого</b>				

### Модуль: Хайтек

№ п/п	Название раздела, темы	Количество часов			Формы аттестации/ контроля
		Всего	Теория	Практика	

	Введение в двумерную графику. Редакторы векторной графики и основные инструменты.				Разработка задания для вырезания
	Устройство и общие принципы работы лазерного станка. Возможные риски при работе с лазерным станком. Техника безопасности.				Участие в обсуждении, выполнение задания практикума
	Трехмерное моделирование. Программы для создания 3D-моделей.				Выполнение задания практикума
	Подготовка модели к производству: программы-слайсеры. Печать изделия.				практикума
	Устройство и общие принципы работы D-принтера. Возможные риски при работе с 3D- принтером.				Обсуждение. Разработка «кодекса безопасности»
<b>Итого:</b>					

#### Формы контроля

Участие во внутренних мероприятиях Технопарка, муниципальных и областных мероприятиях, решение и защита учебных кейсов, защита проекта и создание прототипа или групповые соревнования.

### III. Содержание изучаемого курса

- 1. Введение в образовательную программу, техника безопасности. Первичный инструктаж. (2 часа).**



- *Теория (1 ч.):* Знакомство с группой. Ознакомление учащихся с программой, приемами и формами работы. Первичный инструктаж.
- *Практика (1 ч.):* входное тестирование.

## **2. Кейс 1: «Вычисление периметра и площади заданного объекта». (10 часов).**

- *Теория (2 ч.):* Общие сведения о языке программирования Python. Синтаксис. Ввод и вывод данных. Переменные и типы данных. Базовые алгоритмические конструкции. Стандартные операции с целыми и действительными числами. Стиль программирования Python.
- *Практика (8 ч.):* Структура программы. Решение простых задач в интерактивном режиме.

## **3. Кейс 2: «Обмен значений переменных». (10 часов).**

- *Теория (2 ч.):* Жизненный цикл программного продукта. Основные этапы разработки. Тестирование. Отладка. Основные понятия: логический тип данных, логические операции (and, or, not, A), условный оператор, условное и альтернативное исполнение алгоритма, операторы сравнения, вложенность операторов, оператор цикла, переменная-флаг, генерация псевдослучайной последовательности, инструкции break, continue и pass.
- *Практика (8 ч.):* Решение простых задач в интерактивном режиме.

## **4. Кейс 3: «Определение существования треугольника по трем сторонам». (12 часов).**

- *Теория (4 ч.):* Условная и циклическая инструкции.
- *Практика (8 ч.):* Решение простых задач в интерактивном режиме.

## **5. Кейс 4: «Принадлежит ли точка кругу?». (12 часов).**

- *Теория (4 ч.):* Каскадность и вложенность алгоритмических инструкций.
- *Практика (8 ч.):* Решение простых задач в интерактивном режиме.

## **6. Кейс 5: «Простейший калькулятор». (14 часов).**

- *Теория (4 ч.):* Теория чисел. Фильтрация потока чисел. НОД и НОК. Проверка числа на простоту.
- *Практика (10 ч.):* Решение задач на анализ чисел потока и целочисленной арифметики

## **7. Кейс 6: «Алгоритм Евклида». Повторный инструктаж. (14 часов).**

- *Теория (4 ч.):* Алгоритм Евклида. Нахождение максимума и минимума. Повторный инструктаж.
- *Практика (10 ч.):* Решение задач на анализ чисел потока и целочисленной арифметики повышенной трудности.

## **8. Кейс 7: «Проверка уникальности элементов списка». (14 часов).**

- *Теория (4 ч.):* Основные понятия: список, кортеж, элемент списка и кортежа, индекс, срез списка, матрица, многомерный список, сортировка, сложность алгоритма, устойчивость сортировки, квадратичная, быстрая, синхронная, поразрядная сортировки списка, случайное перемешивание.

- *Практика (10 ч.):* Решение задач на ввод-вывод элементов одномерного и многомерного списка.

#### **9. Кейс 8: «Решето Эратосфена». (14 часов).**

- *Теория (4 ч.):* Представление списка и кортежа в памяти компьютера, сходства и различия. Способы заполнения списка (с клавиатуры, из файла, случайным образом, по формуле). Методы работы со списком и кортежем. Методы сортировки списка.
- *Практика (10 ч.):* Решение задач на ввод-вывод элементов одномерного и многомерного списка и кортежа, преобразование, поиск, замену, подсчет.

#### **10. Кейс 9: «Выбор числа из строки». (14 часов).**

- *Теория (4 ч.):* Методы работы со строкой.
- *Практика (10 ч.):* Применение методов строки. Решение задач.

#### **11. Кейс 10: «Подсчет количества одинаковых элементов списка». (14 часов).**

- *Теория (4 ч.):* Основные понятия: подпрограмма, функция, процедура, рекурсия, глубина рекурсии, объявление, определение и вызов функции, возврат значений, глобальные и локальные переменные, передача параметров, работа памятью, граф вызовов, стек вызовов, полиморфизм функций, утиная типизация, lambda-функции.
- *Практика (10 ч.):* Нахождение суммы чисел. Числа Фибоначчи. Вычисление степени. Ханойские башни. Использование библиотеки математических функций. Решение задач повышенной трудности.

#### **12. Заключительное занятие. (2 часа).**

- *Теория (1 ч.):* Оценка результатов. Повторение.
- *Практика (1 ч.):* Демонстрация лучших решений кейсов.

#### **13. Мероприятия, направленные на развитие общекультурных компетенций. (12 часов).**

- *Теория (4 ч.):* новые знания и теоретические задания по освоению общекультурных компетенций.
- *Практика (8 ч.):* выполнение практических задания по освоению общекультурных компетенций.

### **Модуль Хайтек**

**Теория (6 ч.):** Знакомство с принципами создания векторного графического изображения, изучение инструментария векторного графического редактора. Использование векторного изображения как управляющей программы для лазерного станка. Изучение принципов работы лазерного станка и возможности его использования в практической деятельности.

Изучение основ трехмерного моделирования для последующего создания объектов сложных форм. Подготовка модели к производству с использованием аддитивных технологий. Знакомство с

оборудованием для производства объемных объектов сложных форм, изучение принципов его функционирования, принципиальных отличий технологий.

**Практика (12 ч.):** Освоение методов создания векторных изображений и подготовки задания для лазерной обработки различных материалов – резки, нанесения изображения (гравировка), получение практического опыта применения лазерных технологий при решении функциональных задач.

Освоение специализированного программного обеспечения подготовки модели к печати и управления работой 3D-принтера, основ 3D-моделирования.

#### **IV. КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ**

1.1. Кейсы (Приложение 1).

1.2. Календарный учебный график (Приложение 2).

### 1.3. Ресурсное обеспечение программы:

Рекомендуемое учебное оборудование (на группу из 12 учащихся):

#### **Материально-техническое обеспечение:**

##### **Аппаратное обеспечение:**

- кабинет, оснащенный компьютерной техникой, не менее 1 ПК на 2 ученика.
- проектор;
- доска;
- Доступ к сети Интернет;
- принтер цветной.

##### **Программное обеспечение:**

- ПО Python3;
- операционная система Windows 10 professional;
- браузер google chrome;

**Рекомендуемое учебное оборудование для модуля «Хайтек»  
рассчитанное на группу из 12 учащихся).**

<b>Основное оборудование и материалы</b>	<b>Кол-во</b>	<b>Ед. изм</b>
Компьютер		шт.
3D принтер учебный (Picaso 3D Designer)	1	шт.
3D принтер учебный (Picaso 3D Designer PRO)	1	шт.
D принтер учебный с большой областью печати (Hercules)	1	шт.
D принтер промышленный (Дельта)	1	шт.
D принтер фотополимерный		шт.
D сканер ручной		шт.
Лазерный станок Trotec	1	шт.
Принтер цветной (A4 / A3)		шт.
Плоттер		шт.
Пластик для 3D принтеров и ручек		кг.
Фанера (не ниже 3 сорта) 4 мм		лист
Оргстекло (2 мм/ 4 мм/ 8 мм)		лист
Проектор		шт.
Экран		шт.
Набор инструментов для постобработки (наждачная бумага, надфили и др.)		набор

<b>Дополнительное оборудование и материалы</b>	<b>Кол.</b>	<b>Ед. изм.</b>
Вышивальная машина		шт.
Пылесос		шт.
Мусорный бак (большой)		шт.

#### **Информационно-методическое обеспечение**

Основной организационной формой обучения в ходе реализации данной образовательной программы является занятие. Эта форма обеспечивает организационную чёткость и непрерывность процесса обучения. Знание педагогом индивидуальных особенностей воспитанников позволяет

эффективно использовать стимулирующее влияние коллектива на учебную деятельность каждого обучающегося.

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

**Формы занятий:** соревнования, выставки, конкурсы, практикум, занятие – консультация, занятие - ролевая игра, занятие – презентация, занятие проверки и коррекции знаний и умений.

Методы организации учебного процесса:

- Информационно – рецептивный метод (предъявление педагогом информации и организация восприятия, осознания и запоминание учащимися данной информации).
- Репродуктивный метод (составление и предъявление педагогом заданий на воспроизведение знаний и способов умственной и практической деятельности, руководство и контроль за выполнением; воспроизведение воспитанниками знаний и способов действий по образцам, произвольное и произвольное запоминание).
- Метод проблемного изложения (постановка педагогом проблемы и раскрытие доказательно пути его решения; восприятие и осознание обучающимися знаний, мысленное прогнозирование, запоминание).
- Эвристический метод (постановка педагогом проблемы, планирование и руководство деятельности учащихся; самостоятельное решение обучающимися части задания, произвольное запоминание и воспроизведение).
- Исследовательский метод (составление и предъявление педагогом проблемных задач и контроль за ходом решения; самостоятельное планирование обучающимися этапов, способ исследования, самоконтроль, произвольное запоминание).

В организации учебной познавательной деятельности используются также словесные, наглядные и практические методы.

**Словесные методы.** Словесные методы педагог применяет тогда, когда главным источником усвоения знаний обучающимися является слово (без опоры на наглядные способы и практическую работу). К ним относятся: рассказ, опрос, объяснение и т.д.

**Наглядные методы.** К ним относятся методы обучения с использованием наглядных пособий.

**Практические методы.** Методы, связанные с процессом формирования и совершенствования умений и навыков обучающихся. Основным методом является практическое занятие.

#### 1. **Дидактические средства.**

В ходе реализации образовательной программы педагогом используются дидактические средства: учебные наглядные пособия, демонстрационные устройства, технические средства.

#### 2. **Формы подведения итогов:** промежуточные проекты, тестирования

Формы и методы обучения:

1. Формирование и совершенствование умений и навыков (изучение нового материала, практика).
2. Познавательный (восприятие, осмысление и запоминание учащимися нового материала с привлечением наблюдения готовых примеров, моделирования, изучения иллюстраций, восприятия, анализа и обобщения демонстрируемых материалов).
3. Метод проектов (при усвоении и творческом применении навыков и умений в процессе разработки собственных моделей).
4. Систематизирующий (опрос по теме, составление систематизирующих таблиц, графиков, схем и т.д.).
5. Контрольный метод (при выявлении качества усвоения знаний, навыков и умений и их коррекция в процессе выполнения практических заданий).
6. Групповая работа (используется при совместной работе над кейсом, а также при разработке проектов).
7. Индивидуальная работа (используется при работе с одарёнными детьми и детьми - инвалидами)

Рефлексия

Возможность обдумать то, что учащиеся запрограммировали, помогает им более глубоко понять идеи, с которыми они сталкиваются в процессе своей деятельности на предыдущих этапах. Размышляя, учащиеся устанавливают связи между полученной ими новой информацией и уже знакомыми им идеями, а также предыдущим опытом.

Развитие

Творческие задачи, представляющие собой адекватный вызов способностям ребёнка, наилучшим образом способствуют его дальнейшему обучению и развитию. Радость свершения, атмосфера успеха, ощущение хорошо выполненного дела – всё это вызывает желание продолжать и совершенствовать свою работу.

### **Формы отслеживания и фиксации результатов**

В течение учебного года для определения уровня усвоения программы учащимися осуществляются диагностические срезы:

- входная диагностика – тестирование, где выясняется стартовый уровень ЗУН учащегося (*Приложение 3*).
- промежуточная диагностика позволяет выявить достигнутый на данном этапе уровень ЗУН учащихся, в соответствии с пройденным материалом программы. Предлагается тестирование,

а также учитывается участие в соревнованиях и проектная деятельность учащихся (*Приложение 4*).

- итоговая диагностика проводится в конце учебного года (демонстрация и публикация проектов) и предполагает комплексную проверку образовательных результатов в виде теста по всем ключевым направлениям, а также учитывается участие в соревнованиях и проектная деятельность учащихся. Данный контроль позволяет проанализировать степень усвоения программы учащимися (*Приложение 5*).

Педагог фиксирует деятельность и результаты учащихся в сводную таблицу результатов обучения (*Приложение 6*).

Итоговые результаты контроля фиксируются в диагностической карте (*Приложение 7*).

#### Виды контроля

Виды контроля	Содержание	Методы	Сроки контроля
Входной	Начальный уровень подготовки учащихся, имеющиеся знания, умения и навыки, связанные с предстоящей деятельностью.	Тестирование	Сентябрь
Промежуточный	Освоение учебного материала за полугодие, позволяет выявить достигнутый на данном этапе уровень ЗУН учащихся, в соответствии с пройденным материалом программы	Тестирование, Демонстрация решений кейса	Декабрь
Текущий	Проверка усвоения знаний	Беседа	В течении года
Итоговый	Проектная деятельность Освоение учебного материала за учебный год, предполагает комплексную проверку образовательных результатов по всем ключевым направлениям	Тестирование, Демонстрация решений кейса	Май

### Оценка уровней освоения программы

Уровни /%	Параметры	Показатели
<b>Высокий уровень/</b>	Теоретические знания.	Оценка теоретических знаний на основе тестирования. Учащийся освоил материал в полном объеме. Знает и понимает значение терминов, самостоятельно ориентируется в содержании материала по темам.
	Практические умения.	Способен свободно применять в практической работе полученные знания. Учащийся проявляет устойчивое внимание к выполнению заданий, сосредоточен во время практической работы, получает результат своевременно. Может оценить результаты выполнения своего задания и дать оценку работы своего товарища.
	Навыки ведения проектной деятельности.	Учащийся прекрасно работает со всеми членами команды. Всегда справляется с поставленной задачей в группе. Свободно генерирует идеи. Легко применяет полученные знания и умения в решении поставленной задачи.
<b>Средний уровень/</b>	Теоретические знания.	Оценка теоретических знаний на основе тестирования. Учащийся освоил базовые знания, но слабо ориентируется в содержании материала по некоторым темам.
	Практические умения.	Владеет базовыми навыками и умениями, но не всегда может в полном объеме выполнить практическое самостоятельное задание, затрудняется и просит помощи педагога. В работе допускает небрежность, делает ошибки, но может устранить их после наводящих вопросов или



		<p>самостоятельно. Оценить результаты своей деятельности может с подсказкой педагога.</p> <p>Учащийся заинтересован, но не всегда проявляет устойчивое внимание к выполнению задания.</p>
	<p>Навыки ведения проектной деятельности.</p>	<p>Учащийся слабо сосредоточен во время работы в группе, не всегда умеет находить общий язык с членами команды. Справляется с поставленной задачей в группе, но просит помощи и подсказки педагога. Не всегда умеет генерировать идеи. Применяет полученные знания и умения в решении поставленной задачи, но с некоторыми подсказками педагога или товарищей.</p>
<b>Низкий уровень/</b>	<p>Теоретические знания.</p>	<p>Оценка теоретических знаний на основе тестирования.</p> <p>Владеет минимальными знаниями, слабо ориентируется в содержании материала.</p>
	<p>Практические умения.</p>	<p>Учащийся способен выполнять каждую операцию практической работы только с подсказкой педагога или товарищей. Не всегда правильно применяет в практической работе необходимые знания или не использует вовсе. В работе допускает грубые ошибки, не может их найти даже после указания. Не способен самостоятельно оценить результаты своей работы.</p>
	<p>Навыки ведения проектной деятельности.</p>	<p>Учащийся слабо контактирует в работе с членами команды. Не умеет генерировать идеи. Не всегда умеет справиться с поставленной задачей в группе. Решение задачи происходит исключительно с подсказкой педагога. Слабо применяет полученные знания и умения в решении поставленной задачи, исключительно с подсказками педагога или товарищей.</p>

## V. Список литературы

### Литература для преподавателя:

1. Arduino Diecimila [Электронный ресурс]/ URL: <http://arduino.ru/Hardware/ArduinoBoardDiecimila> (дата обращения: 17.05.2020).
2. Arduino Programming for Visual Studio and Atmel Studio/ Arduino for Visual Studio [Электронный ресурс]/ URL: <http://www.visualmicro.com/> (дата обращения: 12.04.2020).
3. Sweigart, А. Разработка компьютерных игр на языке Python / А. Sweigart. - 2-е изд., испр. - М. : Национальный Открытый Университет «ИНТУИТ», 2016. - 505 с. : ил. ; То же [Электронный ресурс]. - URL: [//biblioclub.ru/index.php?page=book&id=429009](http://biblioclub.ru/index.php?page=book&id=429009) (28.05.2020).
4. Sweigart, А. Разработка компьютерных игр с помощью Python и Pygame / А. Sweigart. - 2-е изд., испр. - М. : Национальный Открытый Университет «ИНТУИТ», 2016. - 290 с. : ил. ; То же [Электронный ресурс]. - URL: [//biblioclub.ru/index.php?page=book&id=429009](http://biblioclub.ru/index.php?page=book&id=429009) (28.05.2020).
5. Брага Н.С. Создание роботов в домашних условиях. / Н.С.Брага. – М.: НТ Пресс, 2007. – 368 с.
6. Буйначев, С.К. Основы программирования на языке Python : учебное пособие / С.К. Буйначев, Н.Ю. Боклаг ; Министерство образования и науки Российской Федерации, Уральский федеральный университет имени первого Президента России Б. Н. Ельцина. - Екатеринбург : Издательство Уральского университета, 2014. - 92 с. : табл., ил. - Библиогр. в кн. - ISBN 978-5-7996-1198-9 ; То же [Электронный ресурс]. - URL: [//biblioclub.ru/index.php?page=book&id=429184](http://biblioclub.ru/index.php?page=book&id=429184)
7. Вандервурд Д., Джосаттис Н. М. Шаблоны C++: Справочник разработчика / Пер. с англ. – М.: Изд. дом «Вильямс», 2003. – 544 с., ил.
8. Васильев Е.А. Микроконтроллеры. Разработка встраиваемых приложений. – СПб.:БХВ-Петербург, 2008. – 304 с.
9. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. Приемы объектно-ориентированного проектирования. Паттерны проектирования. — СПб.: Питер, 2001.
10. Златопольский, Д. М. Сборник задач по программированию / Д. М. Златопольский.– 2-е изд.– Санкт-Петербург : БХВ-Петербург, 2007.– 240 с.
11. Кнут Д. Э. Искусство программирования. Т. 1. Основные алгоритмы. 3-е изд. / Пер. с англ. – М.: Изд. дом «Вильямс», 2001. – 720 с., ил.
12. Массимо Б. Arduino для начинающих волшебников. / Пер. с англ. под ред. М. Райтман. – М.: Рид Групп, 2012. – 128 с.
13. Предко М. 123 эксперимента по робототехнике. / пер. с англ. Попова В.П. – М.: НТ-Пресс, 2007. – 544с.
14. Предко М. Руководство по микроконтроллерам. Том 1. / Пер. с англ. под ред.И. И. Шагурина и С.Б. Лужанского - М.: Постмаркет, 2001. – 416 с.
15. Ревич, Ю.В. Практическое программирование микроконтроллеров Atmel AVR на языке ассемблер [Текст] / СПб. БХВ-Петербург, 2014-368с.
16. Северенс, Ч. Введение в программирование на Python / Ч. Северенс. - 2-е изд., испр. - М. : Национальный Открытый Университет «ИНТУИТ», 2016. - 231 с. : схем., ил. ; То же [Электронный ресурс]. - URL: [//biblioclub.ru/index.php?page=book&id=429184](http://biblioclub.ru/index.php?page=book&id=429184)
17. Соммер У. Программирование микросхемных плат Arduino/Freduino. – СПб.: БХВ-Петербург, 2012. – 256 с.
18. Стивен Прата Язык программирования C++. Лекции и упражнения, 5-е издание / М.: Вильямс, 2007. – 1248 с.
19. Сузи, Р.А. Язык программирования Python : курс / Р.А. Сузи. - 2-е изд., испр. - М. : Интернет-Университет Информационных Технологий, 2007. - 327 с. - (Основы

информационных технологий). - ISBN 978-5-9556-0109-0 ; То же [Электронный ресурс].

20. Харви Дейтел, Пол Дейтел Как программировать на C++: Третье издание. Пер с англ. - М.: ЗАО "Издательство БИНОМ", 2003. – 1011 с.
21. Хахаев, И.А. Практикум по алгоритмизации и программированию на Python : курс / И.А. Хахаев. - 2-е изд., исправ. - М. : Национальный Открытый Университет «ИНТУИТ», 2016. - 179 с. : ил. - Библиогр. в кн. ; То же [Электронный ресурс]. - URL:

#### **Литература и информационные ресурсы для учащихся:**

1. Массимо Б. Arduino для начинающих волшебников. / Пер. с англ. под ред. М. Райтман. – М.: Рид Групп, 2012. – 128 с.
2. Предко М. 123 эксперимента по робототехнике. / пер. с англ. Попова В.П. – М.: НТ-Пресс, 2007. – 544с.
3. Харви Дейтел, Пол Дейтел Как программировать на C++: Третье издание. Пер с англ. - М.: ЗАО "Издательство БИНОМ", 2003. – 1011 с.
4. Хахаев, И.А. Практикум по алгоритмизации и программированию на Python : курс / И.А. Хахаев. - 2-е изд., исправ. - М. : Национальный Открытый Университет «ИНТУИТ», 2016. - 179 с. : ил. - Библиогр. в кн. ; То же [Электронный ресурс]. - URL:
5. Python: основы и применение
6. Программирование на Python
7. Язык программирования Python
8. Практикум по алгоритмизации и программированию на Python
9. Разработка компьютерных игр с помощью Python и Pygame
10. Введение в программирование на Python

## **Кейсы**

### **Кейс 1: «Вычисление периметра и площади заданного объекта».**

Площадь прямоугольного треугольника равна половине площади прямоугольника, стороны которого равны длинам катетов.

Периметр находится путем сложения длин всех сторон треугольника. Поскольку известны только катеты, гипотенуза вычисляется по теореме Пифагора:

Чтобы извлечь квадратный корень в Python, можно воспользоваться функцией `sqrt()` из модуля `math`.

Найдите площадь и периметр прямоугольного треугольника по двум заданным катетам. Создайте программу.

### **Кейс 2: «Обмен значений переменных».**

Обмен значений двух переменных - это действие, в результате которого одна переменная принимает значение, равное второй переменной, а вторая - первой.

Если попытаться сделать такой обмен, сначала присвоив первой переменной значение второй, а второй - значение первой, ничего не получится. Допустим, есть две переменные `a` и `b`. При этом  $a = 5$  и  $b = 6$ . Если выполнить выражение `a = b`, то переменная `a` будет хранить число 6, также, как и `b`. Число 5 будет утеряно. После этого выражение `b = a` теряет смысл, т. к. `b` будет присвоено его же текущее значение. В данном случае 6.

Поэтому в программировании при обмене значений переменных обычно вводят третью переменную, играющую роль буфера (ее иногда называют буферной переменной). В этой переменной сохраняют значение первой переменной, потом первой переменной присваивают значение второй, а новое значение для второй переменной берут из буфера.

Отобразите алгоритм обмена значений двух переменных на языке Python.

### **Кейс 3: «Определение существования треугольника по трем сторонам».**

У треугольника сумма любых двух сторон должна быть больше третьей. Иначе две стороны просто "лягут" на третью и треугольника не получится.

Пользователь вводит длины трех сторон. Программа должна определять, может ли существовать треугольник при таких длинах. Это значит, необходимо сравнить суммы всех пар сторон с оставшейся третьей стороной. Чтобы треугольник существовал, сумма всегда должна

быть больше отдельной стороны или, по крайней мере, не меньше, если учитывать так называемый вырожденный треугольник.

Поскольку всего три стороны, то можно составить три варианта сложения двух сторон:  $a + b$ ,  $b + c$ ,  $a + c$ . Первую сумму сравниваем с оставшейся стороной  $c$ , вторую - с  $a$  и третью - с  $b$ . Если хотя бы в одном случае сумма окажется не больше третьей стороны, то делается вывод, что треугольник не существует.

Решите поставленную задачу на языке Python.

#### **Кейс 4: «Принадлежит ли точка кругу?».**

Определить, принадлежит ли точка с координатами  $(x; y)$  кругу радиуса  $R$  с центром в начале координат.

Пользователь вводит координаты точки и радиус круга.

Если выбрать точку на координатной плоскости, то можно увидеть, что проекции ее координат на оси  $x$  и  $y$  являются катетами прямоугольного треугольника. А гипотенуза этого прямоугольного треугольника как раз показывает расстояние от начала координат до точки. Таким образом, если длина гипотенузы будет меньше радиуса круга, то точка будет принадлежать кругу; иначе она будет находится за его пределами.

Длину гипотенузы вычисляется по теореме Пифагора: квадрат гипотенузы равен сумме квадратов катетов. Откуда гипотенуза равна квадратному корню из суммы квадратов катетов.

Обратите внимание, можно вводить отрицательные координаты. При возведении в квадрат все-равно будет получено положительное число.

Решите поставленную задачу на языке Python.

#### **Кейс 5: «Простейший калькулятор».**

Написать программу, которая выполняет над двумя вещественными числами одну из четырех арифметических операций (сложение, вычитание, умножение или деление). Программа должна завершаться только по желанию пользователя.

Чтобы программа самопроизвольно не завершалась, в ней надо запустить бесконечный цикл. Выход из него будем осуществлять с помощью оператора `break`, если пользователь вводит определенный символ вместо знака арифметической операции.

Если пользователь ввел знак, который не является ни знаком арифметической операции, ни символом-"прерывателем" работы программы, то вывести сообщение о некорректном вводе.

Если был введен один из четырех знаков операции, запросить ввод двух чисел.

В зависимости от знака операции выполнить соответствующее арифметическое действие.

Если было выбрано деление, необходимо проверить не является ли нулем второе число. Если это так, то сообщить о невозможности деления.

Решите поставленную задачу на языке Python.

### **Кейс 6: «Алгоритм Евклида».**

Алгоритм Евклида – это алгоритм нахождения наибольшего общего делителя (НОД) пары целых чисел.

Наибольший общий делитель (НОД) – это число, которое делит без остатка два числа и делится само без остатка на любой другой делитель данных двух чисел. Проще говоря, это самое большое число, на которое можно без остатка разделить два числа, для которых ищется НОД.

#### *Алгоритм нахождения НОД делением*

Большее число делим на меньшее.

Если делится без остатка, то меньшее число и есть НОД (следует выйти из цикла).

Если есть остаток, то большее число заменяем на остаток от деления.

Переходим к пункту 1.

Пример:

Найти НОД для 30 и 18.

$$30 / 18 = 1 \text{ (остаток 12)}$$

$$18 / 12 = 1 \text{ (остаток 6)}$$

$$12 / 6 = 2 \text{ (остаток 0)}$$

Конец: НОД – это делитель 6.

$$\text{НОД}(30, 18) = 6$$

В цикле в переменную *a* или *b* записывается остаток от деления. Цикл завершается, когда хотя бы одна из переменных равна нулю. Это значит, что другая содержит НОД. Однако какая именно, мы не знаем. Поэтому для НОД находим сумму этих переменных. Поскольку в одной из переменных ноль, он не оказывает влияние на результат.

#### *Алгоритм нахождения НОД вычитанием*

Из большего числа вычитаем меньшее.

Если получается 0, то значит, что числа равны друг другу и являются НОД (следует выйти из цикла).

Если результат вычитания не равен 0, то большее число заменяем на результат вычитания.

Переходим к пункту 1.

Пример:

Найти НОД для 30 и 18.

= 6

$12 - 6 = 6$

$6 - 6 = 0$

Конец: НОД – это уменьшаемое или вычитаемое.

НОД (30, 18) = 6

Блок-схема алгоритма Евклида



Примечание. В модуле `math` языка программирования Python есть функция `gcd()`, вычисляющая наибольший общий делитель двух чисел.

Решите поставленную задачу на языке Python.

### Кейс 7: «Проверка уникальности элементов списка».

В списке чисел проверить, все ли элементы являются уникальными, т. е. каждое число встречается только один раз.

Решить данную задачу на языке Python необходимо несколькими способами. Классический вариант - брать по очереди элементы списка и сравнить каждый со стоящими за

ним. При первом же совпадении элементов делается вывод, что в списке есть одинаковые элементы и работа программы завершается.

Еще одним способом решения может быть использование типа данных "множества" (set). Как известно, в множествах не может быть одинаковых элементов. При преобразовании списка во множество в нем одинаковые элементы будут представлены единожды, то есть дубли удалятся. Если после этого сравнить длину исходного списка и множества, то станет ясно, есть ли в списке одинаковые элементы. Если длины совпадают, значит все элементы списка уникальны. Если нет, значит, были одинаковые элементы.

### **Кейс 8: «Решето Эратосфена».**

Решето Эратосфена – это алгоритм нахождения простых чисел до заданного натурального числа путем постепенного отсеивания составных чисел. Образно говоря, через решето Эратосфена в процессе его тряски проскакивают составные числа, а простые остаются в решете.

Чтобы понять данный алгоритм, вспомним, что числа являются простыми, если делятся только на единицу и самих себя. Первое простое число - это 2, второе простое число - это 3. Теперь начнем рассуждать:

- Все четные числа, кроме двойки, - составные, т. е. не являются простыми, так как делятся не только на себя и единицу, а также еще на 2.
- Все числа кратные трем, кроме самой тройки, - составные, так как делятся не только на самих себя и единицу, а также еще на 3.
- Число 4 уже вышло из игры, так как делится на 2.
- Число 5 простое, так как его не делит ни один простой делитель, стоящий до него.
- Если число не делится ни на одно простое число, стоящее до него, значит оно не будет делиться ни на одно сложное число, стоящее до него.

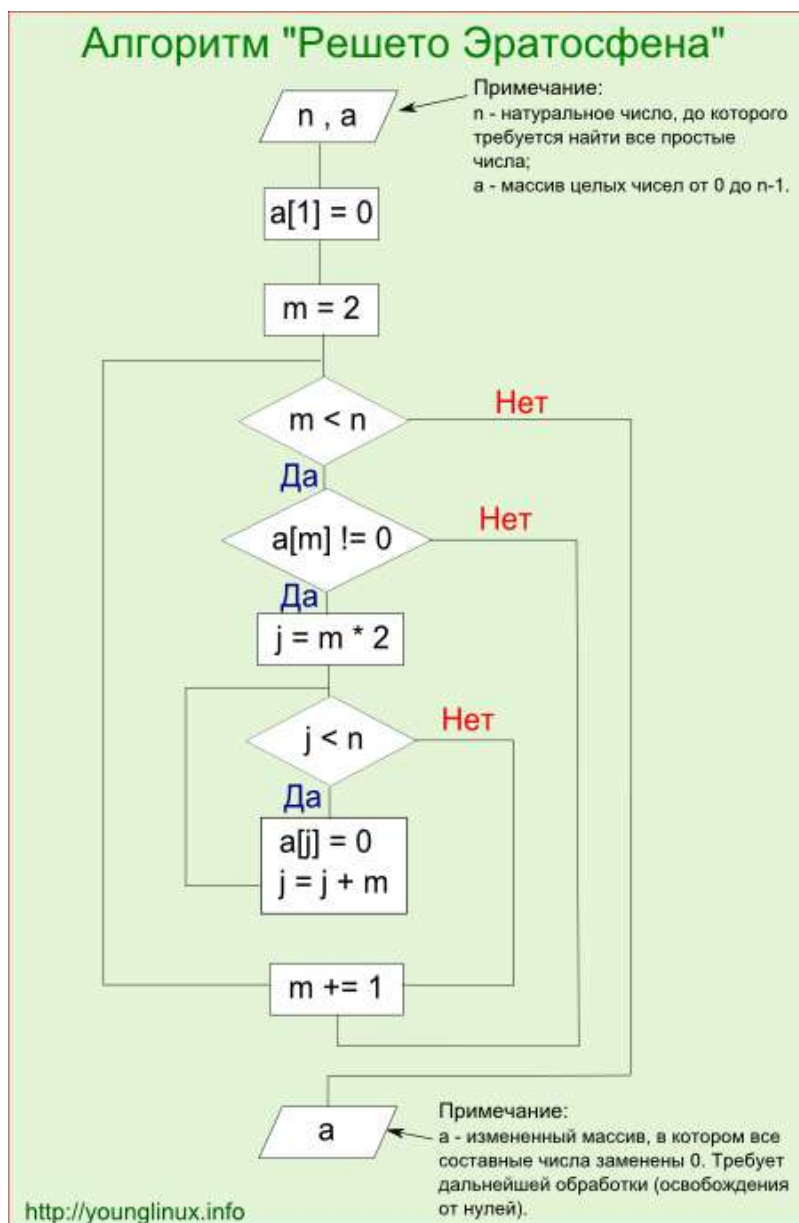
Последний пункт вытекает из того, что сложные числа всегда можно представить, как произведение простых. Поэтому если одно сложное число делится на другое сложное, то первое должно делиться на делители второго. Например, 12 делится на 6, делителями которого являются 2 и 3. Число 12 делится и на 2, и на 3.

Алгоритм Эратосфена как раз заключается в последовательной проверке делимости чисел на предстоящие простые числа. Сначала берется первое простое и из ряда натуральных чисел высеиваются все кратные ему. Затем берется следующее простое и отсеиваются все кратные ему и так далее.



При реализации алгоритма Эратосфена на языке программирования есть некоторая сложность. Допустим, мы помещаем натуральные числа до заданного числа  $n$  в массив. Далее в процессе выполнения алгоритма будем заменять обнаруженные сложные числа нулями. После выполнения алгоритма те ячейки массива, которые не содержат нули, содержат простые числа, которые выводятся на экран.

Однако индексация массива начинается с нуля, а простые числа начинаются с двойки. Эта проблема решаема, но добавляет сложности в код. Поскольку алгоритм Эратосфена не такой уж простой, легче пренебречь началом и взять массив от 0 до  $n$ . Здесь важнее индексы, чем значения элементов. Значениями может быть True, обозначающее простое число, и False, обозначающее сложное число.



**Кейс 9: «Выбор числа из строки».**

Дана строка, содержащая натуральные числа и слова. Необходимо сформировать список из чисел, содержащихся в этой строке. Например, задана строка "abc83 cde7 1 b 24". На выходе мы должны получить список [83, 7, 1, 24].

Следует посимвольно перебирать строку. Если очередной символ цифра, надо добавить ее в новую строку. Далее проверять символы за ней, и если они тоже цифры, то добавлять их в конец этой новой подстроки из цифр. Когда очередной символ окажется не цифрой, или будет достигнут конец строки, то надо преобразовать строку из цифр в число и добавить в список.

Решите поставленную задачу на языке Python.

### **Кейс 10: «Подсчет количества одинаковых элементов списка».**

Дан список целых чисел. Посчитать, сколько раз в нем встречается каждое число. Например, если дан список [1, 1, 3, 2, 1, 3, 4], то в нем число 1 встречается три раза, число 3 - два раза, числа 2 и 4 - по одному разу.

Для хранения количества каждого встречающегося в списке значения создадим словарь. В нем ключами будут числа, которые встречаются в списке, а значениями - количества этих чисел в списке. Для примера, приведенного выше, в итоге должен был бы получиться такой словарь: {1:3, 3:2, 2:1, 4:1}.

Пусть в программе будет функция, которая заполняет список случайными числами в диапазоне и количестве, указанными пользователем.

Другая функция будет считать количество каждого значения и заносить данные в словарь. Алгоритм подсчета заключается в следующем. Если очередной элемент списка уже есть в качестве ключа словаря, то следует увеличить значение этого ключа на единицу. Если очередного элемента списка нет в качестве ключа в словаре, то такой ключ следует добавить и присвоить ему значение, равное единице.

Для того, чтобы вывести содержимое словаря в отсортированном по возрастанию ключей виде, используется функция sorted(). Она сортирует ключи словаря и помещает их в список.

Решите поставленную задачу на языке Python.

## ***Приложение 2***

**Календарный учебный график на 2021/2022 учебный год  
программы «Введение в разработку программ и компьютерных игр» Линия 2**

Педагог д/о – Патрикеева Ольга Николаевна

Период обучения – 1 год

Кол-во учебных недель - 36

К

Режим проведения занятий: 2 раза в неделю по 2 часа (45 минут)

Праздничные и выходные дни (по производственному календарю при шестидневной рабочей неделе): 01.05.2021, 09-10.05.2021.

Каникулярный период:

В период школьных каникул время занятий в объединении может быть изменено.

с

№	Месяц	Число	Время проведения занятия	Форма занятия	Кол-во часов	Тема занятия	Место проведения	Форма контроля
1.				Л/ПР		Введение в образовательный модуль, техника безопасности. Первичный инструктаж.	каб.205	Беседа. Входная диагностика
2.				Л/ПР		Кейс 1: «Вычисление периметра и площади заданного объекта».	каб.205	Беседа
3.				Л/ПР		Кейс 1: «Вычисление периметра и площади заданного объекта».	каб.205	Беседа
4.				Л/ПР		Кейс 1: «Вычисление периметра и площади заданного объекта».	каб.205	Беседа
5.				Л/ПР		Кейс 1: «Вычисление периметра и площади заданного объекта».	каб.205	Беседа
6.				Л/ПР		Кейс 1: «Вычисление периметра и площади заданного объекта».	каб.205	Демонстрация решений кейса
7.				Л/ПР		Кейс 2: «Обмен значений переменных».	каб.205	Беседа
8.				Л/ПР		Кейс 2: «Обмен значений переменных».	каб.205	Беседа
9.				Л/ПР		Кейс 2: «Обмен значений переменных».	каб.205	Беседа

10.				Л/ПР	Кейс 2: «Обмен значений переменных».	каб.205	Беседа
11.				Л/ПР	Кейс 2: «Обмен значений переменных».	каб.205	Демонстрация решений кейса
12.				Л/ПР	Кейс 3: «Определение существования треугольника по трем сторонам».	каб.205	Беседа
13.				Л/ПР	Кейс 3: «Определение существования треугольника по трем сторонам».	каб.205	Беседа
14.				Л/ПР	Кейс 3: «Определение существования треугольника по трем сторонам».	каб.205	Беседа
15.				Л/ПР	Кейс 3: «Определение существования треугольника по трем сторонам».	каб.205	Беседа
16.				Л/ПР	Кейс 3: «Определение существования треугольника по трем сторонам».	каб.205	Беседа
17.				Л/ПР	Кейс 3: «Определение существования треугольника по трем сторонам».	каб.205	Демонстрация решений кейса
18.				Л/ПР	Мероприятия, направленные на развитие общекультурных компетенций.	каб.205	Беседа
19.				Л/ПР	Мероприятия, направленные на развитие общекультурных компетенций.	каб.205	Беседа
20.				Л/ПР	Мероприятия, направленные на развитие общекультурных компетенций.	каб.205	Демонстрация решений кейса

21.			Л/ПР	Кейс 4: «Принадлежит ли точка кругу?».	каб.205	Беседа
22.			Л/ПР	Кейс 4: «Принадлежит ли точка кругу?».	каб.205	Беседа
23.			Л/ПР	Кейс 4: «Принадлежит ли точка кругу?».	каб.205	Беседа
24.			Л/ПР	Кейс 4: «Принадлежит ли точка кругу?».	каб.205	Беседа
25.			Л/ПР	Кейс 4: «Принадлежит ли точка кругу?».	каб.205	Беседа
26.			Л/ПР	Кейс 4: «Принадлежит ли точка кругу?».	каб.205	Демонстрация решений кейса
27.			Л/ПР	Кейс 5: «Простейший калькулятор».	каб.205	Беседа
28.			Л/ПР	Кейс 5: «Простейший калькулятор».	каб.205	Беседа
29.			Л/ПР	Кейс 5: «Простейший калькулятор».	каб.205	Беседа
30.			Л/ПР	Кейс 5: «Простейший калькулятор».	каб.205	Беседа
31.			Л/ПР	Кейс 5: «Простейший калькулятор».	каб.205	Беседа
32.			Л/ПР	Кейс 5: «Простейший калькулятор».	каб.205	Беседа
33.			Л/ПР	Кейс 5: «Простейший калькулятор».	каб.205	Демонстрация решений кейса
34.			Л/ПР	Кейс 6: «Алгоритм Евклида». Повторный инструктаж.	каб.205	Беседа Промежуточная диагностика
35.			Л/ПР	Кейс 6: «Алгоритм Евклида».	каб.205	Беседа
36.			Л/ПР	Кейс 6: «Алгоритм Евклида».	каб.205	Беседа
37.			Л/ПР	Кейс 6: «Алгоритм Евклида».	каб.205	Беседа

38.				Л/ПР	Кейс 6: «Алгоритм Евклида».	каб.205	Беседа
39.				Л/ПР	Кейс 6: «Алгоритм Евклида».	каб.205	Беседа
40.				Л/ПР	Кейс 6: «Алгоритм Евклида».	каб.205	Демонстрация решений кейса
41.				Л/ПР	Мероприятия, направленные на развитие общекультурных компетенций.	каб.205	Беседа
42.				Л/ПР	Мероприятия, направленные на развитие общекультурных компетенций.	каб.205	Беседа
43.				Л/ПР	Мероприятия, направленные на развитие общекультурных компетенций.	каб.205	Демонстрация решений кейса
44.				Л/ПР	Кейс 7: «Проверка уникальности элементов списка».	каб.205	Беседа
45.				Л/ПР	Кейс 7: «Проверка уникальности элементов списка».	каб.205	Беседа
46.				Л/ПР	Кейс 7: «Проверка уникальности элементов списка».	каб.205	Беседа
47.				Л/ПР	Кейс 7: «Проверка уникальности элементов списка».	каб.205	Беседа
48.				Л/ПР	Кейс 7: «Проверка уникальности элементов списка».	каб.205	Беседа
49.				Л/ПР	Кейс 7: «Проверка уникальности элементов списка».	каб.205	Беседа
50.				Л/ПР	Кейс 7: «Проверка уникальности элементов списка».	каб.205	Демонстрация решений кейса
51.				Л/ПР	Кейс 8: «Решето Эратосфена».	каб.205	Беседа

52.				Л/ПР	Кейс 8: «Решето Эратосфена».	каб.205	Беседа
53.				Л/ПР	Кейс 8: «Решето Эратосфена».	каб.205	Беседа
54.				Л/ПР	Кейс 8: «Решето Эратосфена».	каб.205	Беседа
55.				Л/ПР	Кейс 8: «Решето Эратосфена».	каб.205	Беседа
56.				Л/ПР	Кейс 8: «Решето Эратосфена».	каб.205	Беседа
57.				Л/ПР	Кейс 8: «Решето Эратосфена».	каб.205	Демонстрация решений кейса
58.				Л/ПР	Кейс 9: «Выбор числа из строки».	каб.205	Беседа
59.				Л/ПР	Кейс 9: «Выбор числа из строки».	каб.205	Беседа
60.				Л/ПР	Кейс 9: «Выбор числа из строки».	каб.205	Беседа
61.				Л/ПР	Кейс 9: «Выбор числа из строки».	каб.205	Беседа
62.				Л/ПР	Кейс 9: «Выбор числа из строки».	каб.205	Беседа
63.				Л/ПР	Кейс 9: «Выбор числа из строки».	каб.205	Беседа
64.				Л/ПР	Кейс 9: «Выбор числа из строки».	каб.205	Демонстрация решений кейса
65.				Л/ПР	Кейс 10: «Подсчет количества одинаковых элементов списка».	каб.205	Беседа
66.				Л/ПР	Кейс 10: «Подсчет количества одинаковых элементов списка».	каб.205	Беседа
67.				Л/ПР	Кейс 10: «Подсчет количества одинаковых элементов списка».	каб.205	Беседа
68.				Л/ПР	Кейс 10: «Подсчет количества одинаковых элементов списка».	каб.205	Беседа
69.				Л/ПР	Кейс 10: «Подсчет количества	каб.205	Беседа

					одинаковых элементов списка».		
70.				Л/ПР	Кейс 10: «Подсчет количества одинаковых элементов списка».	каб.205	Беседа. Итоговая диагностика
71.				Л/ПР	Кейс 10: «Подсчет количества одинаковых элементов списка».	каб.205	Демонстрация решений кейса
72.				Л/ПР	Заключительное занятие	каб.205	Беседа. Итоговая диагностика



**Входная диагностика.**

Тестирование.

Ф.И. \_\_\_\_\_

**Вопрос 1**

Какие имена являются правильными в PYTHON

**Варианты ответов**

- G
- ABC
- 41N
- Game2
- a+b
- \_ab

**Вопрос 2**

Какие существуют типы переменных (выбрать несколько вариантов):

**Варианты ответов**

- float
- str
- num
- int
- bool
- real

**Вопрос 3**

Переменная float:

**Варианты ответов**

- вещественная переменная
- символьная строка
- логическая переменная
- целая переменная

**Вопрос 4**

Переменная int:

**Варианты ответов**

- вещественная переменная
- символьная строка
- логическая переменная
- целая переменная

**Вопрос 5**

Переменная bool:

#### Варианты ответов

- вещественная переменная
- символьная строка
- логическая переменная
- целая переменная

#### *Вопрос 6*

Переменная str:

#### Варианты ответов

- вещественная переменная
- символьная строка
- логическая переменная
- целая переменная

#### *Вопрос 7*

Имена переменных не могут включать:

#### Варианты ответов

- Русские буквы
- Латинские буквы
- Пробелы
- Скобки, знаки + = ! ? b др.
- Цифры

**Промежуточная диагностика.**

Тестирование.

Ф.И. \_\_\_\_\_

**Вопрос 1**

Что будет в результате выполнения программы:

```
a = 20
```

```
b = a + 4
```

```
a = b * 100
```

```
print(a)
```

В качестве ответа введите число \_\_\_\_\_

**Вопрос 2**

Что будет в результате следующего действия `print(2**3)`

В качестве ответа введите число \_\_\_\_\_

**Вопрос 3**

Что будет в результате выполнения следующего действия `print(23 % 2)`

Введите число: \_\_\_\_\_

**Вопрос 4**

Результатом вычисления `print(23 // 3)` будет число: \_\_\_\_\_

**Вопрос 5**

Что будет в результате выполнения следующего алгоритма:

Входные данные: 57

```
x = int(input())
```

```
if x > 0:
```

```
    print(x)
```

```
else:
```

```
    print(-x)
```

\_\_\_\_\_

**Вопрос 6**

Что будет в результате выполнения следующего алгоритма:

Входные данные: -57

```
x = int(input())
if x > 0:
    print(x)
else:
    print(-x)
```

---

### *Вопрос 7*

Что будет в результате выполнения следующего алгоритма программы:

Входные данные:

15

45

```
a = int(input())
b = int(input())
if a % 10 == 0 or b % 10 == 0:
    print('YES')
else:
    print('NO')
```

**Варианты ответов**

- YES
- NO

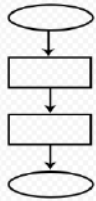
**Итоговая диагностика.**

Тестирование.

Ф.И. \_\_\_\_\_

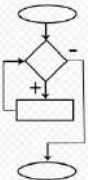
1. Язык программирования Python подходит для разработки:
  - a. Компьютерных и мобильных приложений
  - b. Аналитика и машинное обучение
  - c. Игр
  - d. Ничего из этого.

2. Назовите тип алгоритма:



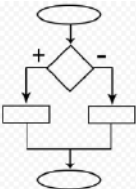
- a. Разветвляющийся
- b. Линейный
- c. Циклический
- d. Смешанный

3. Назовите тип алгоритма:



- a. Линейный
- b. Разветвляющийся
- c. Циклический
- d. Смешанный

4. Назовите тип алгоритма:



- a. Разветвляющийся
- b. Линейный
- c. Смешанный
- d. Циклический

5. Что хранит в себе переменная?

- a. Имя
  - b. Значение
  - c. Тип
  - d. Длину своего значения
6. Что обозначает тип данных int?
- a. Целочисленное
  - b. Вещественное
  - c. Строковое
  - d. Булевое
7. Выберите правильную запись оператора присваивания:
- b.  $x = 7,8$
  - c.  $a = 5$
  - d.  $a == b + x$
8. Укажите оператор ввода:
- a. input()
  - b. print()
  - c. int()
  - d. random()
9. Сколько возможных значений у переменной типа bool?
- a. 2
  - b. 4
  - c. 10
  - d. Сколько угодно
10. Какой оператор здесь используется?
- ```
If n < 100:  
    b = n + a
```
- a. Условный оператор
  - b. Оператор присваивания
  - c. Оператор сложения
  - d. Оператор умножения
11. Что лучше использовать для множественного ветвления?
- a. if – elif – else
  - b. Много if
  - c. if – else – elif
  - d. while
12. Оператор цикла в языке Python:
- a. while
  - b. for
  - c. if
  - d. print
13. Сколько раз произойдет итерация цикла? (Итерация – единичное выполнение тела цикла)

```
print("Осталось", total)
```

```
total = 100  
i = 0  
while i < 0:  
    n = int(input())  
    total = total - n  
    i = i + 1
```

- a. 4
- b. 5
- c. 6
- d. 0

14. Для чего нужен оператор `break`?

- a. Для завершения программы
- b. Для выхода из цикла
- c. Для поломки компьютера
- d. Для удаления программы

15. Где находятся параметры, а где аргументы функции?

- a. Параметры пишутся при объявлении функции, аргументы при вызове
- b. Аргументы пишутся при объявлении функции, параметры при вызове
- c. Это одно и то же!
- d. У функции есть только параметры

16. Что делает функция `len()`?

- a. Возвращает длину строки
- b. Возвращает случайное число
- c. Возвращает номер символа
- d. Возвращает модуль числа

17. Как добавить модуль в программу?

- a. `import math`
- b. `import math()`
- c. `import (math)`
- d. `import.math`

18. На каких операционных системах может работать Python?

- a. Windows
- b. Linux
- c. macOS
- d. Ничего из этого

19. От чего язык программирования называется «Питон»?

- a. В честь змеи
- b. В честь ТВ-шоу
- c. В честь игры
- d. В честь блюда

20. Сколько уже лет языку программирования Python?

- a. 30
- b. 1
- d. 1

21. Создатель языка программирования Python

- a. Гвидо Ван Россум
- b. Дэвид Паттерсон
- c. Эрвин Дональд Кнут
- d. Джеймс Артур Гослинг

22.  $a = 345$ . Что выведет команда `print(//100)`

- a. 3
- b. 5
- c. 4
- d. 34

23. Выберите циклический алгоритм

- a. `k = 0`  
`while k < 10:`  
    `print("Привет")`  
    `k += 1`
- b. `a = int(input())`  
`b = int(input())`  
`c = int(input())`  
`s = a+b+c`  
`print(c)`
- c. `a = int(input())`  
`if a > 0:`  
    `print(a)`  
`else:`  
    `print(a)`

24. В какой строке правильно записан ввод числа с клавиатуры?

- 1. `a = int(input())`
- 2. `b=input(int())`
- 3. `c=int(input())`
- 4. `s=a+b+c`
- 5. `print(s)`

- a. 1
- b. 2
- c. 3
- d. 4
- e. 5

25. В какой строке допущена ошибка?

- 1. `a =`  
    `int(input())`
- 2. `if a>0:`
- 3.     `print(a):`
- 4. `else`
- 5.     `print(A)`

- a. 1
- b. 2



- c. 5
- d. 4

26. Что такое «else»?

- a. Так как
- b. Иначе
- c. Если
- d. Потому что

27. Сколько раз программа напишет слово «Пока»?

```
k=0
while k<10:

    print(«Привет»)
    k += 1
```

- a. 9
- b. 0
- c. 10
- d. Бесконечно

ОТВЕТЫ:

- 1. a,b,c
- 2. b
- 3. c
- 4. a
- 5. b
- 6. a
- 7. c
- 8. a
- 9. a
- 10. a,b,c
- 11. a
- 12. a,b
- 13. b
- 14. b
- 15. a
- 16. a
- 17. a
- 18. a,b,c
- 19. b
- 20. a
- 21. a
- 22. a
- 23. a
- 24. c
- 25. c,d
- 26. b
- 27. b



**Сводная таблица результатов обучения**  
учащихся по дополнительной общеобразовательной программе  
**Введение в разработку программ и компьютерных игр» Линия 2**

педагог д/о \_\_\_\_\_

Группа № \_\_\_\_\_ год обучения \_\_\_\_\_

| <b>№п/п</b>   | <b>ФИ<br/>учащегося</b> | <b>Оценка<br/>теоретических<br/>знаний<br/>(на основе<br/>тестирования)</b> | <b>Участие в<br/>соревнованиях<br/>(10 баллов за<br/>каждое)</b> | <b>Работа над<br/>проектом<br/>(максимум 5 баллов<br/>за каждый проект)</b> | <b>Средний<br/>балл</b> | <b>Процент</b> |
|---------------|-------------------------|-----------------------------------------------------------------------------|------------------------------------------------------------------|-----------------------------------------------------------------------------|-------------------------|----------------|
| 1.            |                         |                                                                             |                                                                  |                                                                             |                         |                |
| 2.            |                         |                                                                             |                                                                  |                                                                             |                         |                |
| 3.            |                         |                                                                             |                                                                  |                                                                             |                         |                |
| 4.            |                         |                                                                             |                                                                  |                                                                             |                         |                |
| 5.            |                         |                                                                             |                                                                  |                                                                             |                         |                |
| 6.            |                         |                                                                             |                                                                  |                                                                             |                         |                |
| 7.            |                         |                                                                             |                                                                  |                                                                             |                         |                |
| 8.            |                         |                                                                             |                                                                  |                                                                             |                         |                |
| 9.            |                         |                                                                             |                                                                  |                                                                             |                         |                |
| <b>Итого:</b> |                         |                                                                             |                                                                  |                                                                             |                         |                |

**Диагностическая карта**

учащихся по дополнительной общеобразовательной программе

**Введение в разработку программ и компьютерных игр» Линия 2**

Педагог д/о \_\_\_\_\_

Группа № \_\_\_\_\_ год обучения \_\_\_\_\_

Вид контроля \_\_\_\_\_

| <b>№ п/п</b>  | <b>ФИ учащегося</b> | <b>Уровень освоения программы</b> |
|---------------|---------------------|-----------------------------------|
| <b>1.</b>     |                     |                                   |
| <b>2.</b>     |                     |                                   |
| <b>3.</b>     |                     |                                   |
| <b>4.</b>     |                     |                                   |
| <b>5.</b>     |                     |                                   |
| <b>6.</b>     |                     |                                   |
| <b>7.</b>     |                     |                                   |
| <b>8.</b>     |                     |                                   |
| <b>9.</b>     |                     |                                   |
| <b>Итого:</b> |                     |                                   |