МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Департамент образования Ямало-Ненецкого автономного округа

Департамент образования Администрации МО г.Салехард

МАОУ «Лицей»

«Рассмотрено» Руководителем кафедры дополнительного образования _____ О.Г. Шабардина Протокол №1 от «30» августа 2023г	**«Согласовано»** Председатель НМС _____ Е.С.Высоцкая Протокол № 1 от «31» августа 2023 г.	**«Утверждено»** Директор _____ Н.И.Решетняк Приказ № 529-а от «31» августа 2023 г.

ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

Название	ПРОГРАМИРОВАНИЕ НА ЯЗЫКЕ PYTHON
Класс	7-11 классы
Учебный год	2023
Срок реализации программы	4 месяца
Учитель (ФИО)	Черепенин А.Ю.

г. Салехард
2023 г.

1. НОРМАТИВНЫЕ ДОКУМЕНТЫ

1.	Федеральный закон от 29.12.2012 №273-ФЗ « Об образовании в Российской Федерации» (п. 22 ст. 2, ч. 1; ст.12, ч. 7 ст. 28, ст. 30, п. 5 ч. 3 ст. 47, п. 1 ч. 1 ст. 48).
2.	Приказ Министерства просвещения Российской Федерации от 27.07.2022 № 629 «Об утверждении Порядка организации осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
3.	Постановление Главного государственного санитарного врача Российской Федерации от 28 сентября 2020 №28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно – эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодёжи»;
4.	Постановление Главного государственного санитарного врача Российской Федерации 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»;
5	Приказ Министерства труда и социальной защиты Российской Федерации от 22.09.2021 «652н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых»;
6.	Примерная (авторская) программа по объединению дополнительного образования

2. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеобразовательная (общеразвивающая) программа «Программирование на языке Python. Предподготовка» имеет **техническую направленность.** В современную жизнь человека все шире внедряются компьютеры и информационные технологии. Поэтому все большее значение приобретает компьютерная грамотность. Курс изучения компьютерной грамотности состоит из двух разделов: пользовательского курса и программирования.

Актуальность раздела «Программирование» в школьном курсе представлена языком программирования Pascal, а многим учащимся хочется познакомиться с другими языками программирования, самим попробовать разработать программы, которые можно использовать на уроках и во внеурочной деятельности. Данная Программа позволяет реализовать эти желания, так как уделяется большое внимание практической работе учащихся на компьютере, самостоятельной разработке ими программ для решения практических задач.

Новизна Программы заключается в том, что Python дает более широкие возможности в области программирования, чем Pascal, который входит в школьный курс информатики. На языке Python можно легко и быстро создавать простые компьютерные игры, трёхмерные модели и программировать роботов. Этот язык быстрее и легче усваивается, чем Pascal. Многие мировые компании такие, как Intel, Cisco, Hewlett-Packard, используют этот язык при реализации своих проектов. Крупнейшие интернет-ресурсы такие, как Google, YouTube, также разработаны с помощью языка программирования Python.

С развитием современных информационных технологий сегодня любой учащийся под руководством опытного педагога может с лёгкостью научиться программировать. Компьютеры и компьютерные системы – неотъемлемая часть жизни нашего общества. Научившись программировать, мы можем быть не только пользователями информационных технологий, но и активными их создателями. Языки программирования можно сравнить с иностранными языками, овладеть ими может каждый. Учиться программировать очень интересно. Результат программирования очень часто виден сразу. Кроме того, создание 4 компьютерных игр и обучающих программ способствует развитию логики и креативного мышления. Ещё одной

значимой стороной обучения программированию является спрос на рынке труда на специалистов данного направления деятельности.

ЦЕЛЬ ПРОГРАММЫ:

обучение учащихся программированию посредством языка Python, развитие инженерного мышления, воспитание конкурентно способной личности.

ЗАДАЧИ ПРОГРАММЫ:

Обучающие:
• обучить языку программирования Python и созданиюпрограмм на его основе;
• научить создавать прикладное программное обеспечение;
• расширять кругозор обучающихся в области программирования;
• научить дизайнерскому оформлению созданного ПО.
Развивающие:
• развивать память и внимание, познавательную и творческую активность;
• развивать творческие способности, эстетическое и эргономическое восприятие объектов труда;
• развивать логическое мышление.
Воспитательные:
• прививать интерес к активному творческому самовыражению, культуре труда;
• воспитывать упорство в достижении желаемого результата;
• воспитывать эстетический вкус;
• воспитывать чувство взаимопомощи, доверия, коллективизма.
Ожидаемые результаты и способы определения их результативности
Учащиеся научатся:
• программированию на языке Python;
• основам дизайнерского оформления созданных программ.
Учащиеся получат возможность научиться:
• производить чтение и запись программ на языке Python;
• запускать и отлаживать программу.

ПРОГРАММА ПОСТРОЕНА С СОБЛЮДЕНИЕМ СЛЕДУЮЩИХ ПРИНЦИПОВ:
• добровольность;
• общественная направленность;
• учет возрастных и индивидуальных особенностей обучающихся;
• предоставление самостоятельности и опора на инициативу;
• учет интересов ребенка;
• содружество и сотворчество детей и взрослых.

Формы подведения итогов реализации программы

Освоение данной программы сопровождается процедурой промежуточной аттестации учащихся проводимой в формах, определенных программой, и учебно-тематическим планом.

Текущий контроль за реализацией программы проводится в разных формах:

- тестирование;
- устный опрос;
- самостоятельные и контрольные работы;
- участие в проектной деятельности.

Адресат программы

Программа рассчитана на учеников в возрасте от 12 до 17 лет, без ОВЗ.

Условия реализации программы

Режим занятий. Программа предполагает 4 месяца (октябрь-январь) обучения. **Продолжительность занятия составляет** 1 час. Занятия проводятся с периодичностью 1 раза в неделю по 1 час. Общее количество часов в неделю – 1 часов, всего – 18 часов.

Планируемые результаты освоения программы

Личностные:

— сформированы навыки разработки индивидуального проекта от составления ТЗ до практической реализации;

— сформированы навыки реализации творческой составляющей во время проектирования дизайна мобильного приложения;

— сформированы навыки командной работы и взаимоуважения;

— разработка проекта требует гибкости и логичности принятых решений;

— сформирован устойчивый интерес к дальнейшему развитию в сфере информационных технологий;

Развивающие:

— развита творческая активность;

— развита познавательная активность.

Социальные:

— сформировано умение пользоваться приемами коллективного творчества;

— сформировано умение эстетического восприятия мира и доброе отношение к окружающим.

Регулятивные:

— умеет соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата;

— умеет определять способы действий в рамках предложенных условий и требований, корректировать свои действия в соответствии с изменяющейся ситуацией.

Познавательные:

— умеет работать с литературой и другими источниками информации;

— умеет самостоятельно определять цели своего обучения.

Коммуникативные:

— умеет организовать учебное сотрудничество и совместную деятельность с педагогом и сверстниками;

— работать индивидуально и в группе, уметь вступать в контакт со сверстниками.

Предметные:

— знание необходимой терминологии («информация», «алгоритм», «исполнитель», «модель»), смысла этих понятий и умение применять полученные знания на практике;

— развитие у обучающихся алгоритмического (знакомство и навыки работы с простейшими алгоритмическими структурами – линейной, условной и циклической), а также логического мышления, что необходимо для грамотного составления алгоритмов, рассчитанных для конкретного исполнителя;

— знание основных понятий и этапов проектной деятельности;

— навыки пошагового выполнения алгоритмов управления исполнителями и анализа числовых и текстовых данных, умение осуществлять данные операции как вручную, так и с использованием компьютера;

— знания основ программирования и областей применения полученных навыков.

Метапредметные:

— ориентироваться в своей системе знаний: отличать новое знание от известного;

— перерабатывать полученную информацию: делать выводы в результате совместной работы группы, сравнивать и группировать предметы и их образы;

— работать по предложенным инструкциям и самостоятельно;

— излагать мысли в четкой логической последовательности, отстаивать свою

точку зрения, анализировать ситуацию и самостоятельно находить ответы на вопросы путем логических рассуждений;

— определять и формировать цель деятельности на занятии с помощью учителя;
— работать в группе и коллективе;
— уметь рассказывать о проекте;
— работать над проектом в команде,
— эффективно распределять обязанности;
— работать над проектом индивидуально, эффективно распределять время.

Критерии оценки

Работа в команде	Самостоятельность выполнении задач	Участие в конкурсах	Специальные знания	Алгоритмическое, логическое мышление	Поиск и обработка информации в сети Интернет
3 балла – умеет вести конструктивный диалог, эффективно распределять роли в команде с целью получения результата, доброжелателенко всем участникам команды. 2 балла – неумеет рационально распределять роли в команде, при общении с членами команды возникают недопонимания 1 балл - бывает нетерпимым к	3 балла-справляется с заданием самостоятельно, возможны подсказки со стороны педагога. 2 балла-справляется самостоятельно невсегда, прибегает к помощи педагога. 1 балл–без помощи в большинстве случаев справитсяс заданием не может.	3 балла– Участие вразного вида конкурсах ипроектах, неоднократноявляется победителем. 2 балла– Участие вразного вида конкурсах ипроектах, наблюдаютсяразовые победы, входит в 10-ку лучшихучастников по рейтинговой таблице соревнований 1 балл– Участие вразного вида конкурсах ипроектах, входит впервые 50% рейтинговой таблицы соревнований .	3 балла-100% выполнение 2 балла-70% выполнение 1 балл – 50% выполнение	3 балла-не возникает проблем с выявлением причины и следствия определенного решения 2 балла-возникают проблемы с выявлением причины, либо следствия. 1 балл – У учащегося возникают проблемы и с теми с другим.	3 балла-безпроблем ориентируется в просторах интернета, находит информацию позапросу, умеет фильтровать и отсеивать ненужную информацию. 2 балла-ориентируется в просторах интернета, находит информацию позапросу, иногда теряется в информационном потоке. 1 балл – можетнайти информацию в интернете, но подобрать подходящий информационный контент без помощи состороны других участников образовательного процесса не в состоянии, сомневается в правильности выбора информации.

окружающим, не умеет распределять роли в команде.					

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№	Содержание	Кол-во часов			Формы контроля
		Всего	Теория	Практика	
1.	**Введение в Python**	4	2	2	
1.1	Язык Python. Среда программирования. Особенности ввода-вывода	2	1	1	Практическое задание
1.2	Типы данных, операции. Оператор присваивания	2	1	1	Выполнение упражнений
2.	**Алгоритмические инструкции**	9	5	4	
2.1	Условный оператор	3	2	1	Опрос
2.2	Цикл while. Теория чисел	4	2	2	Практическое задание
2.3	Цикл for	2	1	1	Практическое задание
3.	**Строки**	5	3	2	
3.1	Литералы строк	2	1	1	Практическое задание
3.2	Срезы строк	1.5	1	0.5	Практическое задание
3.3	Методы строк	1.5	1	0.5	Практическое задание
	Всего	18	9	9	

Содержание учебного (тематического) плана.

1. Введение в Python Основные понятия: трансляция, интерпретация, компиляция, синтаксис, семантика, прагматика, переменная, динамическая типизация, служебные слова, идентификаторы, простые типы данных, приоритеты операций, литералы чисел, операция присваивания, PEP 8.

1.1. Язык Python. Среда программирования. Особенности вводавывода Теория. Язык программирования Python. Достоинства и недостатки. Области применения. Интерактивный режим работы программы. Практика. Установка языка программирования Python 3.5 и среды программирования WinglDE 100. Регистрация на Интернет-ресурсах.

1.2. Типы данных, операции. Оператор присваивания Теория. Ввод и вывод числовой информации. Практика. Тренировочное задание на ввод и вывод числовой информации.

1.3. Числа. Стандартные операции Теория. Стандартные операции с целыми и действительными числами. Стиль программирования Python (PEP 8). Практика. Решение простых задач в интерактивном режиме.

2. Алгоритмические инструкции 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.

2.1. Условный оператор Теория. Условная и циклическая инструкции. Каскадность и вложенность алгоритмических инструкций.

2.2. Цикл while Теория. Теория чисел. Фильтрация потока чисел. НОД и НОК. Проверка числа на простоту. Алгоритм Евклида. Нахождение максимума и минимума. Практика. Решение задач на анализ чисел потока и целочисленной арифметики.

2.3. Цикл for Теория. Теория чисел. Фильтрация потока чисел. НОД и НОК. Проверка числа на простоту. Алгоритм Евклида. Нахождение максимума и минимума. Практика. Решение задач на анализ чисел потока и целочисленной арифметики повышенной трудности.

3. Строки Основные понятия: символ, строка, литерал, таблицы кодов ASCII, UTF-8, отладка кода, неизменяемый объект, формат вывода строки, экранированные escape- последовательности, положительная и отрицательная нумерация символов в строке, срез, конкатенация, длина строки.

3.1. Литералы строк Теория. Понятие «литералы строк». Практика. Ввод-вывод строки. Решение задач на ввод строки, поиск подстроки.

3.2. Срезы строк Теория. Форматирование строки. Практика. Преобразование строки. Решение упражнений.

3.3. Методы строк Теория. Методы работы со строкой. Практика. Применение методов строки. Решение задач.

4. Функции Основные понятия: подпрограмма, функция, процедура, рекурсия, глубина рекурсии, объявление, определение и вызов функции, возврат значений, глобальные и локальные переменные, передача параметров, работа с памятью, граф вызовов, стек вызовов, полиморфизм функций, утиная типизация, lambda-функции.

4.1. Парадигма и преимущества структурного программирования Теория. Обзор парадигм программирования. Особенности применения языков программирования.

4.2. Граф и стек вызовов функций. Области видимости переменных Теория. Применение стек и граф вызовов функций. Практика. Выполнение тренировочных упражнений.

4.3. Прямая рекурсия Теория. Понятие прямой рекурсии и ее применение. Практика. Выполнение тренировочных упражнений.

4.4. Косвенная рекурсия Теория. Понятие косвенной рекурсии и ее применение. Практика. Выполнение тренировочных упражнений. Практические занятия к темам 4.2.,4.3.,4.4.: Нахождение суммы чисел. Числа Фибоначчи. Вычисление степени. Ханойские башни. Использование библиотеки математических функций. Решение задач повышенной трудности.

5. Списки и кортежи 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.

5.1. Списки и кортежи в Python. Сходства и различия Теория. Представление списка и кортежа в памяти компьютера, сходства и различия.

КАЛЕНДАРНО-ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

№	Тема	Кол-во часов	Дата	Коррекция
ВВЕДЕНИЕ В PYTHON (4 ч.)				
1	Язык Python. Среда программирования. Особенности ввода-вывода (теория)	1		
2	Язык Python. Среда программирования. Особенности ввода-вывода (практика)	1		
4	Типы данных, операции. Оператор присваивания (теория)	2		
АЛГОРИТМИЧЕСКИЕ ИНСТРУКЦИИ (9 ч.)				
7	Условный оператор (теория)	2		
8	Цикл while. Теория чисел (теория)	1		
9	Цикл while. Теория чисел (теория)	1		
10	Цикл while. Теория чисел (практика)	2		
16	Цикл for (теория)	2		
17	Цикл for (практика)	1		
СТРОКИ (5 ч.)				
23	Литералы строк (теория)	1		
24	Литералы строк (практика)	1		
25	Литералы строк (теория)	2		
26	Срезы строк (теория)	1		

3. ИНФОРМАЦИОННО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

Список литературы.

Литература, использованная при подготовке программы

1. К. Ю. Поляков, Е. А. Еремин. Информатика. Углублённый уровень. Учебник для 10 класса в 2 частях. М.: БИНОМ. Лаборатория знаний,2014.

2. М. Лутц. Изучаем Python. СПб.: Символ-Плюс,2011.

3. Задачи по программированию. Под ред. С.М.Окулова, М.: БИНОМ. Лаборатория знаний,2006.

4. С. М. Окулов. Основы программирования. М.: Бином. Лаборатория знаний, 2012.

Литература, рекомендованная обучающимся

1. М. Лутц. Изучаем Python. СПб.: Символ-Плюс, 2011. 2. Информатика и ИКТ. Задачник-практикум в 2частях. Под ред. И.Г. Семакина и Е. К. Хеннера. М.: БИНОМ. Лаборатория знаний,2014.