



Правительство Санкт-Петербурга
Комитет по образованию

Государственное бюджетное общеобразовательное учреждение
Лицей №126
Калининского района Санкт-Петербурга

Принята

на педагогическом совете

Протокол № 1 от 30 августа 2024 г.

Утверждаю

Директор _____ А. А. Рагимова

Приказ № 351 от 30 августа 2024 г.

Дополнительная общеразвивающая программа

«Программирование на языке "Python":
"Трудные вопросы программирования"»

возраст обучающихся 13 - 18 лет
срок освоения: 36 недель

Составители программы:
Крылатых Данил Николаевич,
педагог дополнительного образования
Гриненко Александр Андреевич,
педагог дополнительного образования

Санкт-Петербург

2024

I. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Основные характеристики программы

Дополнительная общеразвивающая программа «Программирование на языке "Python": "Трудные вопросы программирования"» (далее - ДООП) имеет **техническую направленность**.

Адресат программы

Дополнительная общеразвивающая программа «Программирование на языке "Python": "Трудные вопросы программирования"» предполагает начальный уровень компьютерной грамотности, к занятиям в группах допускаются лица в возрасте 13 - 18 лет.

Актуальность ДООП

Актуальность программы обусловлена высоким интересом подростков к IT-сфере. Важнейшей характеристикой подрастающего поколения является активность в информационном пространстве, интернет-коммуникации. Python является очень востребованным языком программирования - он отлично подходит для знакомства с различными современными парадигмами программирования и активно применяется в самых разных областях от разработки веб-приложений до машинного обучения. Программа «Основы промышленного программирования» в связи с востребованностью на рынке, является своего рода уникальным образовательным продуктом в области информационных технологий. В условиях динамично развивающейся сферы информационных технологий, потребности кадров, развития инфраструктуры программа соответствует региональным социально-экономическим и социокультурным потребностям и проблемам региона.

Программа разработана в соответствии со следующими документами:

- Федеральный закон от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации», (далее - ФЗ № 273);
- Концепция развития дополнительного образования детей до 2030 года (распоряжение Правительства РФ от 31 марта 2022 г. N 678-р), (далее – Концепция);
- Постановление Главного государственного санитарного врача от 28.09.2020 г. № 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;
- 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» (рзд.VI. Гигиенические нормативы по устройству, содержанию и режиму работы организаций воспитания и обучения, отдыха и оздоровления детей и молодежи»)
- Устав Государственного бюджетного общеобразовательного учреждения Лицей № 126 Калининского района Санкт-Петербурга и другими действующими нормативно-правовыми актами с учетом возрастных и индивидуальных особенностей обучающихся на занятиях.

ДООП ежегодно обновляется с учетом развития науки, техники, культуры, экономики, технологий и социальной сферы.

Отличительная особенность ДООП

Отличительная особенность заключается в том, что программа составлена таким образом, чтобы обучающиеся могли овладеть всем комплексом знаний по организации исследовательской деятельности, выполнению проектной работы, познакомиться с требованиями, предъявляемыми к оформлению и публичному представлению результатов

своего труда, а также приобрести практические навыки работы по созданию социально значимых проектов в виде программы, написанной на языке программирования Python.

Уровень освоения ДОП

Дополнительная общеразвивающая программа «Программирование на языке "Python": "Трудные вопросы программирования"» рассчитана на общекультурный уровень освоения программы.

Объем и срок освоение программы

Общее количество учебных часов, запланированных на весь период обучения: 144 академических часа. Срок реализации программы 36 недель.

Цель и задачи ДОП

Цель программы: создание условий для изучения методов программирования на языке Python; рассмотрение различных парадигм программирования, предлагаемых этим языком (процедурная, функциональная, объектно-ориентированная).

ДООП «Программирование на языке "Python": "Трудные вопросы программирования"» ставит перед собой следующие **задачи:**

Обучающие (предметные):

- познакомить с конструкциями языка программирования Python;
- познакомить с принципами и методами функционального и объектно-ориентированного программирования;
- познакомить с основными структурами данных и типовыми методами обработки этих структур;
- сформировать навыки работы в интегрированной среде разработки на языке Python;
- сформировать навыки разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;
- сформировать и развить навыки алгоритмического и логического мышления, грамотной разработки программ;

Развивающие (метапредметные):

- способствовать развитию навыков проектной деятельности;
- способствовать развитию навыков поиска информации в сети Интернет, анализа выбранной информации на соответствие запросу, использования информации при решении задач;
- сформировать у обучающихся интерес к программированию, самостоятельность и творческий подход к решению задач с использованием средств вычислительной техники;
- способствовать развитию навыков работы с различными источниками информации, необходимой для решения учебных задач;
- познакомить с правилами индивидуального и коллективного безопасного поведения при работе с компьютерной техникой;
- способствовать развитию умения планировать свои действия с учётом фактора времени, в обстановке с элементами конкуренции, предвидеть результат и достигать его, при необходимости вносить коррективы в первоначальный замысел.

Воспитательные (личностные):

- способствовать воспитанию этики групповой работы;
- сформировать коммуникативную культуру обучающихся, как внутри проектных групп, так и в коллективе в целом;
- способствовать воспитанию упорства в достижении результата;
- сформировать целеустремлённость, организованность, неравнодушие,

ответственное отношение к труду и уважительное отношение к окружающим.

Планируемые результаты освоения программы

Личностные:

- развита способность к саморазвитию и самообразованию, любознательность, внимательность и настойчивость при выполнении заданий практического характера;
- сформированы навыки организации учебного сотрудничества и совместной деятельности с учителем и сверстниками в процессе проектной и учебно-исследовательской деятельности;
- сформировано ответственное отношение к учению, готовность и способность обучающихся к саморазвитию и самообразованию, средствами информационных технологий;
- развита коммуникативная компетентность в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности;
- развита база целостного мировоззрения, соответствующего современному уровню развития информационных технологий.

Метапредметные:

- сформированы навыки инженерного мышления, умения работать по предложенным инструкциям, программирования, проектирования и эффективного использования электронного вычислительного оборудования;
- развиты внимательность, аккуратность и изобретательность при работе с техническими устройствами, разработке приложений и выполнении учебных проектов;
- сформированы творческое мышление и воображение, умение излагать мысли в четкой логической последовательности, отстаивать свою точку зрения, грамотно работать с критикой и извлекать из неё пользу, анализировать ситуацию и самостоятельно находить ответы на вопросы путем логических рассуждений и информационного поиска.

Предметные:

- получены навыки определения результата выполнения алгоритма при заданных исходных данных, работы с алгоритмами обработки чисел и числовых последовательностей, создания на их основе несложных программ анализа данных, понимания несложных программ, написанных на выбранном для изучения универсальном алгоритмическом языке;
- сформированы основные предметные понятия («информация», «алгоритм», «исполнитель», «модель») и их свойства;
- сформированы логические способности и алгоритмическое мышление, умение составить и записать алгоритм для конкретного исполнителя;
- развиты навыки выполнения (с использованием компьютера или вручную) алгоритмов управления исполнителями и анализа числовых и текстовых данных;
- получены навыки и опыт разработки программ в выбранной среде программирования, включая тестирование и отладку программ, выполнения созданных программ;
- получен опыт разработки и использования компьютерно-математических моделей, оценки числовых параметров моделируемых объектов и процессов, интерпретации результатов, полученных в ходе моделирования реальных процессов, анализа готовых моделей на предмет соответствия реальному объекту или процессу.

Теоретическая подготовка включает: владение теоретическими знаниями (по основным разделам учебного плана программы), системой понятий; владение специальной терминологией; знание правил и алгоритмов деятельности.

Практическая подготовка ребенка включает: практические умения и навыки, предусмотренные программой; владение специальным оборудованием и оснащением; творческие навыки.

Организационно-педагогические условия реализации программы

Язык реализации программы – образовательная деятельность по дополнительной общеразвивающей программе «Программирование на языке "Python": "Трудные вопросы программирования"» осуществляется на русском языке.

Форма обучения: очная.

Особенности реализации программы: реализация программы проходит в течение всего учебного года, включая каникулярное время, с использованием электронного обучения и дистанционных образовательных технологий.

Особенности организации образовательного процесса: образовательный процесс по реализации ДОП «Программирование на языке "Python": "Трудные вопросы программирования"» строится на принципах научности, доступности, наглядности, систематичности и последовательности, связи теории с практикой, сознательности и активности обучения, индивидуальном подходе в обучении с учётом психолого-педагогических особенностей каждого обучающегося.

Условия набора в коллектив: набор в объединение осуществляется по желанию при отсутствии медицинских противопоказаний.

Условия формирования групп: группы формируются разновозрастные.

Количество детей в группе: 12 человек.

Режим занятий: 2 раза в неделю по 2 часа – 144 академических часа в год.

Формы организации занятий: занятия проводятся по группам, программой предусматриваются как аудиторные, так и внеаудиторные формы организации занятий.

Формы проведения занятий: основной формой организации деятельности является учебное занятие традиционной формы.

Формы организации деятельности учащихся на занятии: фронтальная, индивидуальная, групповая.

Материально-техническое оснащение ДОП:

- проектор и экран для демонстрации учебного материала;
- персональные компьютеры для обучающихся с доступом к сети Интернет;
- операционная система Windows
- среда программирования Python

II. УЧЕБНЫЙ ПЛАН

№ п/п	Название раздела, темы	Кол-во часов			Формы контроля /аттестации
		всего	теория	практика	
1.	Введение в программирование.	26	7	19	Устный опрос, беседа, решение задач
2.	Базовые конструкции в Python	42	6	36	Контрольные задания Учебное тестирование
3.	Решение прикладных задач в Python	76	28	48	Устный опрос, беседа, Практическая работа. Разработка/защита индивидуального/группового проекта.
ИТОГО		144	41	103	

Утверждаю

Директор _____ А. А. Рагимова

Приказ № 351 от 30 августа 2024 г.

III. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

Календарный учебный график реализации дополнительной
общеразвивающей программы «Программирование на языке "Python": "Трудные вопросы
программирования"»
на 2024 - 2025 учебный год

Педагог: Крылатых Данил Николаевич

Год обучения	Дата начала занятий	Дата окончания занятий	Количество учебных недель	Количество во учебных дней	Количество во учебных часов	Режим занятий
1 год	01.09.24	31.05.25	36	72	144	2 раз в неделю по 2 часа

III. РАБОЧАЯ ПРОГРАММА

Содержание программы

ТЕМА 1. Введение в программирование

Теория: Инструктажи по технике безопасности с отметкой в журнале. Понятия кода, интерпретатора, программы. Простейшие программы с выводом на экран. Интегрированные среды, исполнение кода. Основные понятия программирования: исполнитель, система команд, алгоритм, программа, среда разработки. Условный оператор. Переменные и арифметика. Погружение в условия. Ввод-вывод в программе, условный оператор, оператор цикла с предусловием. Простейшие программы с использованием условного оператора if, оператора циклов while и операторов ввода-вывода. Технология разработки программы.

Практика: Знакомство с виртуальной средой взаимодействия – регистрация, организация личного кабинета, поиск и выкладывание материалов; знакомство с системой автоматизированной проверки задач и системой Яндекс. Решение задач.

ТЕМА 2. Базовые конструкции в Python

Теория: Понятие о языке высокого уровня Python. Структура программы, переменные и константы, работа с числовыми переменными, арифметические операторы в Python. Основные управляющие конструкции алгоритмов с ветвлением в Python. Знакомство со срезами и диапазонами. Равенство и совпадение объектов. Устройство циклов for. Основные управляющие конструкции циклического алгоритма в Python. Простейшие циклы и циклы с переменными. Списочные выражения. Методы split и join. Другие методы списков и строк. Знакомство с функциями. Области видимости переменных. Работа со списками, строками, множествами и кортежами в Python. Понятие итератора, подпрограммы, процедуры, функции. Функции в Python.

Практика: Решение задач. Защита мини-проекта.

ТЕМА 3. Решение прикладных задач в Python

Теория: Функции, функциональная парадигма программирования. Понятие лямбда функции и области ее применения. Поточковый ввод sys.stdin, обработка коллекций, работа с потоковым вводом, обработка поточной информации. Использование встроенных библиотек (работа с графикой, аудио файлами, презентациями, текстовыми файлами, таблицами и диаграммами). Причины появления, принципы и основные сущности объектно-ориентированного подхода к разработке ПО. Инкапсуляция, полиморфизм, Основы объектно-ориентированного анализа и дизайна, Восходящий и нисходящий способы разработки. Методика объектно-ориентированного дизайна приложения. Шаблоны проектирования. Разработка классов для игры «Шахматы», создание функции цвета. Разработка классов фигур, реализация сруба фигур, реализация. Tkinter: создание графического интерфейса. Знакомство с библиотекой tkinter, изучение основных команд, виджетов, упаковщиков и приемов работы. Принцип работы итераторов и генераторов, создание собственных итераторов и генерирующих функций Презентация тем итоговых проектных работ.

Практика: Решение задач. Создание приложения с графическим интерфейсом. Выбор проектного задания. Поиск открытых данных по выбранной теме. Распределение ролей внутри проектной команды. Подготовка группового / индивидуального проекта. Защита проекта.

V. МЕТОДИЧЕСКИЕ и ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

5.1. Методические материалы

УМК программы состоит из трех компонентов:

1. Учебные и методические пособия для педагога и учащихся.
2. Система средств обучения.
3. Система средств контроля результативности обучения.

Учебные и методические пособия для педагога и учащихся

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.
2. К. Вордерман и др. Программирование для детей: Иллюстрированное руководство по языкам Scratch и Python, 224 стр. Издательство: Манн, Иванов и Фербер, 2017 г.; 3.
3. Б. Пэйн. Python для детей и родителей, 352 стр. Издательство: Эксмо, 2017
4. Материалы и презентации к урокам в LMS Яндекс. Лицея.

5.2. Оценочные материалы

Для отслеживания результативности образовательной деятельности по программе проводятся: входной, текущий, промежуточный и итоговый.

Входной контроль – оценка стартового уровня образовательных возможностей учащихся при поступлении в объединение, ранее не занимающихся по данной дополнительной общеобразовательной общеразвивающей программе. Входная диагностика проводится в сентябре с целью выявления первоначального уровня знаний и умений.

Входной контроль проводится в форме тестирования.

Текущий контроль – оценка уровня и качества освоения тем программы и личностных качеств учащихся осуществляется на занятиях в течение всего учебного года для отслеживания уровня освоения учебного материала программы и развития ИТ-компетенций. Текущий контроль осуществляется посредством наблюдения и беседы.

Промежуточный контроль – оценка уровня и качества освоения учащимися дополнительной общеобразовательной общеразвивающей программы по итогам изучения раздела.

Итоговый контроль – оценка и качество освоения учащимися дополнительной общеобразовательной общеразвивающей программы по завершению учебного года. Итоговый контроль проходит в форме защиты итоговых проектных работ.

Формы фиксации результатов:

- диагностическая карта определения результатов по программе «Программирование на языке "Python": "Трудные вопросы программирования"» (Приложение №1);
- анкета для родителей «Отношение родительской общественности к качеству образовательных услуг и степень удовлетворенности образовательным процессом в объединении»;
- фотографии защиты проектов.

Утверждаю
Директор _____ А. А. Рагимова

Приказ № 351 от 30 августа 2024 г.

Календарно-тематический план

№ п/п	Тема занятия	Кол-во часов	Дата занятия	
			план	факт
1.	Вводное занятие. Проведение мотивирующих бесед с обучающимися. Инструктаж по технике безопасности.	2	1 неделя	
2.	Знакомство со средой при программировании Python	2	1 неделя	
3.	Команды print(), input()	2	2 неделя	
4.	Условный оператор if	2	2 неделя	
5.	Вложенные условия	2	3 неделя	
6.	Простые встроенные функции	2	3 неделя	
7.	Простые встроенные функции. Обмен значениями переменных	2	4 неделя	
8.	Знакомство с циклом while	2	4 неделя	
9.	Знакомство с циклом while. Примеры решения задач	2	5 неделя	
10.	Отладчик. Типы ошибок. Точка останова	2	5 неделя	
11.	Знакомство с циклом for	2	6 неделя	
12.	True и False, break и continue	2	6 неделя	
13.	Вложенные циклы. Принцип работы	2	7 неделя	
14.	Вложенные циклы. Операторы break и continue	2	7 неделя	
15.	Множества. Операции над множествами	2	8 неделя	
16.	Множества. Сравнение множеств	2	8 неделя	
17.	Строки. Индексация	2	9 неделя	
18.	Строки. Срезы	2	9 неделя	
19.	Знакомство со списками	2	10 неделя	
20.	Списки. Основные методы	2	10 неделя	
21.	Кортежи. Преобразование коллекций	2	11 неделя	
22.	Кортежи. Сортировка пузырьком	2	11 неделя	
23.	Методы split и join. Списочные выражения	2	12 неделя	
24.	Методы списков и строк	2	12 неделя	
25.	Подготовка к самостоятельной работе	2	13 неделя	
26.	Самостоятельная работа на простые коллекции	2	13 неделя	
27.	Вложенные списки. Создание двумерного списка	2	14 неделя	
28.	Вложенные списки. Матрицы	2	14 неделя	
29.	Знакомство со словарями	2	15 неделя	
30.	Методы словарей	2	15 неделя	
31.	Подготовка к контрольной работе №1	2	16 неделя	
32.	Контрольная работа №1	2	16 неделя	
33.	Функции. Основные понятия	2	17 неделя	
34.	Функции. Возвращение значений из функций	2	17 неделя	
35.	Функции. Области видимости переменных	2	18 неделя	
36.	Функции. Передача параметров	2	18 неделя	
37.	Функции. Переменное количество аргументов	2	19 неделя	

38.	Функции как объект. Лямбда-функции	2	19 неделя	
39.	Функции. Рекурсия	2	20 неделя	
40.	Обработка коллекций. Методы all, any	2	20 неделя	
41.	Потоковый ввод sys.stdin	2	21 неделя	
42.	Подготовка к самостоятельной работе	2	21 неделя	
43.	Самостоятельная работа на функции	2	22 неделя	
44.	Библиотеки Python. Встроенные модули	2	22 неделя	
45.	Библиотеки Python. Графика	2	23 неделя	
46.	Библиотеки Python. Звук	2	23 неделя	
47.	Библиотеки Python. Морфология	2	24 неделя	
48.	Библиотеки Python. Документы	2	24 неделя	
49.	Библиотеки Python. Numpy	2	25 неделя	
50.	Подготовка к самостоятельной работе	2	25 неделя	
51.	Самостоятельная работа на работу с библиотеками	2	26 неделя	
52.	Введение в ООП. Основные понятия	2	26 неделя	
53.	Введение в ООП. Методы классов	2	27 неделя	
54.	Введение в ООП. Полиморфизм	2	27 неделя	
55.	Введение в ООП. Проверка типа объекта	2	28 неделя	
56.	ООП. Определение операторов	2	28 неделя	
57.	ООП. Специальные методы	2	29 неделя	
58.	ООП. Наследование	2	29 неделя	
59.	ООП. Иерархия классов	2	30 неделя	
60.	ООП. Переопределение методов	2	30 неделя	
61.	ООП. Множественное наследование	2	31 неделя	
62.	Итераторы и коллекции	2	31 неделя	
63.	Практика по работе с графикой и tkinter	2	32 неделя	
64.	Переопределены функции и декораторы	2	32 неделя	
65.	Проектирование и разработка классов	2	33 неделя	
66.	Проектирование и разработка классов	2	33 неделя	
67.	Работа над собственным проектом	2	34 неделя	
68.	Работа над собственным проектом	2	34 неделя	
69.	Подготовка к контрольной работе №2	2	35 неделя	
70.	Контрольная работа №2	2	35 неделя	
71.	Резервное время	2	36 неделя	
72.	Резервное время	2	36 неделя	

Диагностическая карта определения результатов по программе

за учебный год _____

Группа:

Педагог:

Даты проведения: вход:

1 полугодие:

2 полугодие _____

№	Фамилия	Входной контроль	Промежуточный контроль	Итоговый контроль
1.				
2.				
3.				
4.				
5.				
6.				
7.				
8.				
9.				
10.				
11.				
12.				

Уровень освоения обучающимися программы

Параметры результативности	Входной контроль (количество / %)	Промежуточный контроль (количество / %)	Итоговый контроль (количество / %)
Высокий уровень			
Средний уровень			
Начальный уровень			

Входной контроль

по программе дополнительного образования

«Программирование на языке "Python": "Трудные вопросы программирования"»

ФИО _____

Группа _____

Задача 1. Кот и миска (3 балла)

Пока хозяйки нет дома, кот Барсик двигает свою миску с едой. В первый час отсутствия хозяйки Барсик подвинул миску вправо на 30 см и вперёд на 15 см. Во второй час кот подвинул миску вправо ещё на 10 см и назад на 5 см. В третий час Барсик подвинул миску влево на 5 см и назад на 10 см. На сколько сантиметров переместилась миска от изначального расположения?

Задача 2. Догонялки (4 балла)

Антон и Валера бегут по прямой дороге, при этом скорость Валеры постоянная и Антон пытается догнать Валеру, так как отстаёт от него. Антон посчитал, что если сейчас будет бежать со скоростью 10 км/ч, то догонит Валеру за 4 минуты, а если будет бежать со скоростью 15 км/ч, то догонит Валеру за 2 минуты. Чему равна скорость Валеры?

Задача 3. Стулья (3 балла)

Петя побывал на заводе по производству мебели, где увидел стулья на трёх, четырёх и пяти ножках. Всего Петя насчитал 20 стульев и 76 ножек у них. Сколько было стульев на трёх ножках, если стульев на четырёх ножках было в 2 раза больше, чем стульев на пяти ножках?

Задача 4. Подарки (4 балла)

Маша любит дарить подарки своим близким. Маша за год накопила X рублей и решила часть из накопленного потратить на подарки близким. $1/4$ от накопленного она потратила на подарок маме, $1/3$ от оставшегося после покупки подарка маме она потратила на подарок папе, а $1/2$ от оставшегося после покупки подарков маме и папе она потратила на подарок своему младшему брату. В итоге у Маши осталось 1000 рублей.

Чему равен X ?

Задача 5. Хот-доги (5 баллов)

Андрей, Дима и Влад участвуют в соревнованиях по поеданию хот-догов. Всего каждый из них должен был съесть 20 хот-догов. Когда Диме оставалось съесть ещё 4 хот-дога, Андрей уже успел съесть всё. А когда Владу оставалось съесть ещё 5 хот-догов, Дима уже успел съесть все свои хот-доги. Сколько хот-догов успел съесть Влад, когда Андрей съел 10 хот-догов, если едят они с постоянной скоростью?

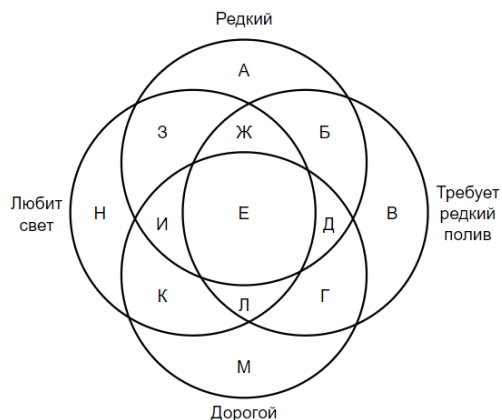
Задача 6. Песочница (2 балла)

Мальши устроили в песочнице гонки на игрушечных машинках. Вася рулил своим игрушечным джипом, у Пети машинка – не скорая помощь, не пожарная, не полиция. Борис выбирает: поехать на кабриолете или взять полицейскую машинку? У Феди есть полицейская машинка и скорая помощь. У Толи есть все машинки: скорая, пожарная, полиция, джип и кабриолет.

Кто на какой машинке участвовал в гонках, если у всех машинки были разные?

Задача 7. Цветок (2 балла)

На приведенной ниже схеме выберите зону, в которой находится цветок, который бы отвечал следующим параметрам: дешевый, редкий цветок, требует редкий полив, не любит свет.



В какой зоне находится такой цветок?

Задача 8. Часы (3 балла)

Какое время должны показать следующие часы?

В ответе запишите четыре цифры времени, опустив двоеточие, например 08:27 - записывается как 0827.



Задача 9. Порядок слов (4 балла)

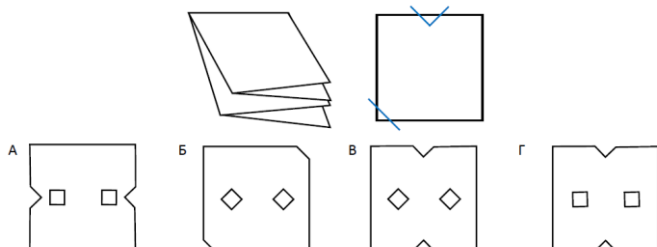
Расположите слова в таком порядке, чтобы любые два соседние слова имели ровно одну общую букву (только одну!) и были при этом разной длины.

Первое слово - Океан.

Океан, дрон, врач, люк, злодей, мох

Задача 10. Лист бумаги (2 балла)

Квадратный листок сложили пополам и ещё раз пополам, как показано на рисунке, а затем отрезали кусочки по синим линиям. Что мы увидим, если развернем сложенный листок (не поворачивая)?



Промежуточный контроль
по программе дополнительного образования
«Программирование на языке "Python": "Трудные вопросы программирования"»

Напишите программы для 3 задач:

1. Внеземная статуэтка (25 баллов)

Напишите программу, убирающую монстров из строк.

Формат ввода

Вводится монстр – слово, которое в любом регистре должно быть в начале хотя бы одного слова выбранной строки.

Затем вводятся строки слов, записанных через пробел.

Формат вывода

Нужно выбрать строки, в которых есть монстр в начале хотя бы одного слова строки (в любом регистре), но не совпадает с ним полностью.

Вывести отобранные строки в исходном порядке.

2. Дом Йохансена (30 баллов)

Напишите функцию для обмена знаниями с единственным выжившим в катастрофе.

Во внешней переменной **house** находится список строк.

Функция **knowledge()** принимает аргумент-строку и именованный аргумент *check* со значением по умолчанию *mate*.

Функция должна найти во внешнем списке **house** строки, в которых есть как минимум две разные буквы из значения именованного аргумента *check*, заменить такие строки на строку-аргумент, а исходные строки в том же порядке переместить в начало списка.

Функция ничего не возвращает, а изменяет внешнюю переменную.

3. Древние со звёзд (45 баллов)

Напишите функцию для защиты от служителей страшных культов.

Функция **ancients()** принимает произвольное число кортежей (строка, число) и произвольное количество именованных аргументов, среди которых могут быть такие:

- *choice* – что выбирать – если аргумент передан, то выбираются строки из тех кортежей, в которых количество общих букв строки и значения аргумента *choice* не меньше числа в этом же кортеже (регистр не учитывать, одинаковые буквы считаются один раз); если аргумент не передан, то строка выбирается из кортежа, если количество использованных в ней букв без учёта регистра не меньше числа этого же кортежа;
- *transform* – во что преобразовать – если аргумент передан, то в нем находится функция для преобразования строк; если не передан, строки не меняются;
- *to_sort* – как сортировать – если аргумент передан и имеет значение *True*, то возвращаемый список строк должен быть отсортирован по возрастанию; если *False*, то по убыванию; если аргумент не передан, список не сортируется.

Функция возвращает список отобранных и преобразованных (а, возможно, и отсортированных) строк.

Критерии оценивания
промежуточного контроля
по программе дополнительного образования
«Программирование на языке "Python": "Трудные вопросы программирования"»

По итогам выполнения группового проекта возможно набрать 100 баллов.

Высокий уровень освоения программы курса - 85 – 100 баллов.

Средний уровень освоения программы курса - 65 – 84 баллов.

Низкий уровень освоения программы курса – 64 и менее баллов.

Итоговый контроль
по программе дополнительного образования
«Программирование на языке "Python": "Трудные вопросы программирования"»

Задание: Реализуйте полноценную игру в шахматы для 2-х человек, которые играют по очереди за одним компьютером.

Таблица 1.

№	Критерии	Мах балл	Оценка
1.	Объём программного кода	15	
2.	Чистота кода (Оценивается PEP 8, константы в коде, сору-paste, говорящие имена переменных и так далее.)	5	
3.	Качество проектирования (Используются ли классы, модули и прочее. Если код написан очень примитивно, то оценка снижается.)	20	
4.	Применённые технологии (Реализованы ли практически все изученные в модуле технологии.)	20	
5.	Работоспособность	5	
6.	Дополнительный балл	5	
		70 баллов	

Критерии оценивания
итогового контроля
по программе дополнительного образования
«Программирование на языке "Python": "Трудные вопросы программирования"»

Высокий уровень освоения программы курса - 55 – 70 баллов.

Средний уровень освоения программы курса - 40 – 54 баллов.

Низкий уровень освоения программы курса – 40 и менее баллов.

Приложение 3.**Перечень интеллектуальных и творческих конкурсов, мероприятий, направленных на развитие интеллектуальных и творческих способностей, интереса к научной (научно-исследовательской), инженерно-технической, изобретательской деятельности**

№ п/п	Название мероприятия	Уровень мероприятия	Возрастная категория	Примерные сроки проведения
1.	НТО	Всероссийский	12 – 18	Сентябрь – ноябрь
2.	«Pythonic Mastery»	Городской	13 – 18	Декабрь
3.	«Codeforces»	Всероссийский	12 – 18	Январь – март
4.	Открытый международный фестиваль по робототехнике и программированию «Red fest»	Региональный	10 – 18	Февраль – апрель

