|  |
| --- |
| Практическая работа 3Работа с циклами в Python **Цель работы:**познакомиться с циклическими конструкциями В Python существуют два типа цикличных выражений:   * Цикл while * Цикл for   **1. Цикл while в Python**     Инструкция while в Python повторяет указанный блок кода до тех пор, пока указанное в цикле логическое выражение будет оставаться истинным. Синтаксис цикла while:  **while логическое выражение**:  команда 1 команда 2 ... команда n      После ключевого слова while указывается условное выражение, и пока это выражение возвращает значение True, будет выполняться блок инструкций, который идет далее.     Все инструкции, которые относятся к циклу while, располагаются на последующих строках и должны иметь отступ от начала строки (4 пробела).  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1545625222444/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-5/while.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-5/while.png?attredirects=0)  Пример программы на Python  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1545625244424/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-5/while_res.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-5/while_res.png?attredirects=0)  Результат выполнения программы с использованием циклического оператора while  **2.Цикл for в Python:**     Цикл for в Python обладает способностью перебирать элементы любого комплексного типа данных (например, строки или списка).      Синтаксис цикла for: **for** int**in**range():     команда 1      команда 2      ...     команда n  Переменной int присваивается значение первого элемента функции range(), после чего выполняются команды. Затем переменной int присваивается следующее по порядку значение и так далее до тех пор, пока не будут перебраны все элементы функции range().  Функция range() является универсальной функцией Python для создания списков (list) содержащих арифметическую прогрессию. Чаще всего она используется в циклах for.  range(старт, стоп, шаг) - так выглядит стандартный вызов функции range() в Python. По умолчанию старт равняется нулю, шаг единице.  **Вариант 0**  1. Найти сумму n элементов следующего ряда чисел: 1 -0.5 0.25 -0.125 ... n. Количество элементов (n) вводится с клавиатуры. Вывести на экран каждый член ряда и его сумму. Решить задачу используя циклическую конструкцию for.  **Решение:**  В данном случае ряд чисел состоит из элементов, где каждый следующий меньше предыдущего в два раза по модулю и имеет обратный знак. Значит, чтобы получить следующий элемент, надо предыдущий разделить на -2.  Какой-либо переменной надо присвоить значение первого элемента ряда (в данном случае это 1). Далее в цикле добавлять ее значение к переменной, в которой накапливается сумма, после чего присваивать ей значение следующего элемента ряда, разделив текущее значение на -2. Цикл должен выполняться n раз.  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547274154746/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-5/ex_for_var0_prog.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-5/ex_for_var0_prog.png?attredirects=0)  Пример программы с циклом for  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547274168171/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-5/ex_for_var0_res.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-5/ex_for_var0_res.png?attredirects=0)  Результат выполнения программы  2. Дано целое число, не меньшее 2. Выведите его наименьший натуральный делитель, отличный от 1.  **Решение:**  Для начала введём целое число командой int(input(текст сообщения)).  Затем зададим переменной i значение 2. Переменная i выполняет роль счётчика. Если задать ей значение 1, то условие задачи не будет выполнено, а результатом всегда будет 1.  В цикле while в качестве логического выражения используется команда n%i сравниваемая с нулём. Таким образом, если остаток от деления введённого числа на текущее значение i не равно нулю, то счётчик увеличивается на 1, а если равно нулю цикл заканчивается и командой print() выводится сообщение и значение i.  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547275785094/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-5/ex_while_var0_prog.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-5/ex_while_var0_prog.png?attredirects=0)  Пример программы с циклом while  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547276081707/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-5/ex_while_var0_res.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-5/ex_while_var0_res.png?attredirects=0)  Результат выполнения программы  **Вариант 1**  1. Дано вещественное число – цена 1 кг конфет. Вывести стоимость 1, 2, … 10 кг конфет. Решить задачу используя циклическую конструкцию for.  2. Дана непустая последовательность целых чисел, оканчивающаяся нулем. Найти: а) сумму всех чисел последовательности; б) количество всех чисел последовательности  Решить задачу используя циклическую конструкцию while.  **Вариант** **2**  1. Даны два числа A и B (A<B). Найти сумму всех целых чисел от А до  В включительно. Решить задачу используя циклическую конструкцию for.  2. Дана последовательность отрицательных целых чисел, оканчивающаяся положительным числом. Найти среднее арифметическое всех чисел последовательности (без учета положительным числа).  Решить задачу используя циклическую конструкцию while.  **Вариант 3**  1. Даны два числа A и B (A<B). Найти сумму квадратов всех целых чисел от А до В включительно. Решить задачу используя циклическую конструкцию for.  2. Дана последовательность из n целых чисел. Первое число в последовательности чётное. Найти сумму всех идущих подряд в начале последовательности чётных чисел. Условный оператор не использовать  Решить задачу используя циклическую конструкцию while.  **Вариант 4**  1. Найти среднее арифметическое всех целых чисел от a до 200 (значения a и b вводятся с клавиатуры; a≤ 200). Решить задачу используя циклическую конструкцию for.  2. Дана последовательность из n вещественных чисел, начинающаяся с положительного числа. Определить, какое количество положительных чисел записано в начале последовательности. Условный оператор не использовать.  Решить задачу используя циклическую конструкцию while.  **Вариант 5**  1. Найти сумму всех целых чисел от a до b (значения a и b вводятся с клавиатуры; b ≥a). Решить задачу используя циклическую конструкцию for.  2.Дано целое число N (> 0), являющееся некоторой степенью числа 2: N = 2^K. Найти целое число K — показатель этой степени.  Решить задачу используя циклическую конструкцию while.  **Вариант 6**  1. Найти сумму квадратов всех целых чисел от a до 50 (значение a вводится с клавиатуры; 0 ≤a≤50). Решить задачу используя циклическую конструкцию for.  2.Дано целое число N (> 1). Найти наименьшее целое число K, при котором выполняется неравенство 5^K > N.  Решить задачу используя циклическую конструкцию while.  **Вариант 7**  1. Дана непустая последовательность целых чисел, оканчивающаяся нулем.  Найти:  а) сумму всех чисел последовательности;  б) количество всех чисел последовательности.  Решить задачу используя циклическую конструкцию for.  2.Дано целое число N (> 1). Найти наибольшее целое число K, при котором выполняется неравенство 2^K > N.  Решить задачу используя циклическую конструкцию while.  **Вариант 8**  1. Дана последовательность из n вещественных чисел. Первое число в последовательности нечетное. Найти сумму всех идущих подряд в начале последовательности нечетных чисел. Условный оператор не использовать. Решить задачу используя циклическую конструкцию for.  2.Дано целое число N (> 0). Используя операции деления нацело и взятия остатка от деления, найти количество и сумму его цифр.  Решить задачу используя циклическую конструкцию while.  **Вариант 9**  1. Среди чисел 1, 4, 9, 16, 25, ... найти первое число, большее n. Решить задачу используя циклическую конструкцию for.  2.Среди чисел 1, 5 10, 16, 23, ... найти первое число, большее n. Условный оператор не использовать.  Решить задачу используя циклическую конструкцию while.  **Вариант 10**  1. Известны оценки по физике каждого из 20 учеников класса. Определить среднюю оценку. Решить задачу используя циклическую конструкцию for.  2.Дано число A (> 1). Вывести наибольшее из целых чисел K, для которых сумма 1 + 1/2 + … + 1/K будет меньше A, и саму эту сумму.  Решить задачу используя циклическую конструкцию while.  **Вариант 11**  1. Известно сопротивление каждого из элементов электрической цепи. Все элементы соединены последовательно. Определить общее сопротивление цепи. Решить задачу используя циклическую конструкцию for.  2.Дано целое число N (> 0). Найти наибольшее целое число K, квадрат которого не превосходит N: K^2 ≤ N. Функцию извлечения квадратного корня не использовать.  Решить задачу используя циклическую конструкцию while.  **Вариант 12**  1. Известны оценки по физике каждого ученика двух классов. Определить среднюю оценку в каждом классе. Количество учащихся в каждом классе одинаковое. Решить задачу используя циклическую конструкцию for.  2. Выведите на экран для числа 2 его степени от 0 до 20  Решить задачу используя циклическую конструкцию while.  **Вариант 13**  1. В области 12 районов. Известны количество жителей (в тысячах человек) и площадь (в км2) каждого района. Определить среднюю плотность населения по области в целом. Решить задачу используя циклическую конструкцию for.  2.Мой богатый дядюшка подарил мне один доллар в мой первый день рождения. В каждый день рождения он удваивал свой подарок и прибавлял к нему столько долларов, сколько лет мне исполнилось. Написать программу, указывающую, к какому дню рождения подарок превысит 100$.  Решить задачу используя циклическую конструкцию while.  **Вариант 14**  1. Одноклеточная амеба каждые 3 часа делится на 2 клетки. Определить, сколько клеток будет через 3, 6, 9, ..., 24 часа, если первоначально была одна амеба. Решить задачу используя циклическую конструкцию for.  2. Вывести таблицу значений функции y = -0.5x + x. Значения аргумента (x) задаются минимумом, максимумом и шагом. Например, если минимум задан как 1, максимум равен 3, а шаг 0.5. То надо вывести на экран изменение x от 1 до 3 с шагом 0.5 (1, 1.5, 2, 2.5, 3) и значения функции (y) при каждом значении x.  Решить задачу используя циклическую конструкцию while.  **Вариант 15**  1. Начав тренировки, лыжник в первый день пробежал 10 км. Каждый следующий день он увеличивал пробег на 10% от пробега предыдущего дня. Определить:  а) пробег лыжника за второй, третий, ..., десятый день тренировок;  б) какой суммарный путь он пробежал за первые 7 дней тренировок.  Решить задачу используя циклическую конструкцию for.  2. Найти сумму и произведение цифр, введенного целого числа. Например, если введено число 325, то сумма его цифр равна 10 (3+2+5), а произведение 30 (3\*2\*5).  Решить задачу используя циклическую конструкцию while. |

### Лабораторная работа 4

### Работа со строками в Python

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Цель работы:**познакомится с методами работы со строками. **Учащийся должен:** **Владеть:**     Навыками составления линейных алгоритмов на языке программирования Python с  использованием строковых данных; **Уметь:**     Применять функции и методы строк при обработке строковых данных; **Знать:**     Операции и методы обработки строк.  Строка — базовый тип представляющий из себя неизменяемую последовательность символов;  str от «string» — «строка».  **Функции и методы работы со строками**   |  |  | | --- | --- | | **Функция или метод** | **Назначение** | | **S1 + S2** | Конкатенация (сложение строк) | | **S1 \* 3** | Повторение строки | | **S[i]** | Обращение по индексу | | **S[i:j:step]** | Извлечение среза | | **len**(S) | Длина строки | | **S.join**(список) | Соединение строк из последовательности str через разделитель,  заданный строкой | | **S1.count**(S[, i, j]) | количество вхождений подстроки s в строку s1. Результатом является число.  Можно указать позицию начала поиска i и окончания поиска *j* | | **S.find**(str, [start],[end]) | Поиск подстроки в строке. Возвращает номер первого вхождения или -1 | | **S.index**(str, [start],[end]) | Поиск подстроки в строке. Возвращает номер первого вхождения или  вызывает ValueError | | **S.rindex**(str, [start],[end]) | Поиск подстроки в строке. Возвращает номер последнего вхождения или  вызывает ValueError | | **S.replace**(шаблон, замена) | Замена шаблона | | **S.split**(символ) | Разбиение строки по разделителю | | **S.upper**() | Преобразование строки к верхнему регистру | | **S.lower**() | Преобразование строки к нижнему регистру |   Ниже приведена программа, демонстрирующая использование функций и методов работы со строками.  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547081137555/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-3/ex_string.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-3/ex_string.png?attredirects=0)  Пример программы на Python  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547081157833/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-3/ex_string_res.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-3/ex_string_res.png?attredirects=0)  Результат выполнения программы с использованием функций и методов работы со строками  **Пример**  **Вариант 0**  Проверить, будет ли строка читаться одинаково справа налево и слева направо (т. е. является ли  она палиндромом).  **Решение**  Сначала введём строку командой: s=input('Введите строку ').  Затем определим логическую переменную  flag и присвоим ей значение 1: flag=1.  Для начала в введённой строке нужно удалить пробелы. Для этого воспользуемся циклической  конструкцией for, которая выполнится столько раз, какую имеет длину строка. Длину строки  определим функцией len(s).  В теле цикла будем проверять следующее условие: s[i]!=' '. Данное логическое выражение будет  истинно в том случае, если i-ый элемент строки не будет равен пробелу, тогда выполнится команда  следующая после двоеточия: string+=s[i].  К сроке string, которая была объявлена в начале программы, будет добавляться посимвольно с  трока s, но уже без пробелов.  Для проверки строки на "палиндром" воспользуемся циклической конструкцией for.  Длина половины строки находится делением нацело на 2.  Если количество символов нечетно,  то стоящий в середине не учитывается, т.к. его сравниваемая пара - он сам.  Количество повторов цикла равно длине половины строки. Длину строки определим функцией  len(s), где аргумент введённая нами строка s. Зная длину строки, можно вычислить количество  повторов цикла. Для этого целочисленно разделим длину строки на 2: len(s)//2.  Для задания диапазона для цикла используем функцию range(), в которой аргументом будет  являться половина длины строки: range(len(s//2 )).  for i in range(len(s//2 )).  Если символ с индексом i не равен "симметричному" символу с конца строки (который находится  путем индексации с конца)  if s[i] != s[-1-i],  то переменной flag присваивается значение 0 и происходит выход из цикла командой break.  Далее, при помощи условной конструкции if-else в зависимости от значения flag либо - 0, либо -1  выводится сообщение, что строка палиндром, либо нет.  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547426930984/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-3/ex_string_var0_prog.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-3/ex_string_var0_prog.png?attredirects=0)  Пример программы на Python  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547427035828/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-3/ex_string_var0_res.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-3/ex_string_var0_res.png?attredirects=0)  Результат выполнения программы  **Задания для самостоятельной работы (по вариантам)**  **Вариант 1**  Дана строка, содержащая русскоязычный текст. Найти количество слов, начинающихся  с буквы "е".  **Вариант 2**  В строке заменить все двоеточия (:) знаком процента (%). Подсчитать количество замен.  **Вариант 3**  В строке удалить символ точку (.) и подсчитать количество удаленных символов.  **Вариант 4**  В строке заменить букву(а) буквой (о). Подсчитать количество замен. Подсчитать, сколько  символов в строке.  **Вариант 5**  В строке заменить все заглавные буквы строчными.  **Вариант 6**  В строке удалить все буквы "а"  и подсчитать количество удаленных символов.  **Вариант 7**  Дана строка. Преобразовать ее, заменив звездочками все буквы "п", встречающиеся среди  первых n/2 символов. Здесь n - длина строки.  **Вариант 8**  Дана строка, заканчивающаяся точкой. Подсчитать, сколько слов в строке.  **Вариант 9**  Определить, сколько раз в тексте встречается заданное слово.  **Вариант 10**  Дана строка-предложение на английском языке. Преобразовать строку так, чтобы каждое  слово начиналось с заглавной буквы.  **Вариант 11**  Дана строка. Подсчитать самую длинную последовательность подряд идущих букв «н».   Преобразовать ее, заменив  точками все восклицательные знаки.  **Вариант 12**  Дана строка. Вывести все слова, оканчивающиеся на букву "я".  **Вариант 13**  Дана строка символов, среди которых есть одна открывающаяся и одна закрывающаяся скобки.  Вывести на экран все символы, расположенные внутри этих скобок.  **Вариант 14**  Дана строка. Вывести все слова, начинающиеся на букву "а" и слова оканчивающиеся на букву "я".  **Вариант 15**   Дана строка текста. Подсчитать количество букв «т» в строке. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лабораторная работа 5Работа со списками. Операции над списками в Python **Цель работы:** Изучение одномерных массивов в Python.  Массивы (списки) в Python — это определенное количество элементов одного типа, которые имеют общее имя, и каждый элемент имеет свой индекс — порядковый номер.  Часто для работы с массивами используются списки.  Список (list) – это структура данных для хранения объектов различных типов.  Списки являются упорядоченными последовательностями, которые состоят из различных типов данных, заключающихся в квадратные скобки [ ] и отделяющиеся друг от друга с помощью запятой.  **Создание списков на Python.**  Создать список можно несколькими способами  **1. Получение списка через присваивание конкретных значений.**  Так выглядит в коде Python пустой список:  s = []  # Пустой список  Примеры создания списков со значениями:  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547171900178/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list_prog.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list_prog.png?attredirects=0)  Списки можно складывать (конкатенировать) с помощью знака «+»:  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547172263718/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list_prog1_res.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list_prog1_res.png?attredirects=0)  Результат:  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547172301822/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list_prog1.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list_prog1.png?attredirects=0)  **2. Создание списка при помощи функции Split().**  Используя функцию split в Python можно получить из строки список.  stroka ="Привет, страна"  lst=stroka.split(",")  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547547457539/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_split_prog.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_split_prog.png?attredirects=0)  Результат:  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547547471226/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_split_res.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_split_res.png?attredirects=0)  **3. Генераторы списков.**  В Python создать список можно также при помощи генераторов.  **Первый способ.**  Сложение одинаковых списков заменяется умножением:  Список из 10 элементов, заполненный единицами  l = [1]\*10  **Второй способ.**  **Пример 1.**  l = [i for i in range(10)]  **Пример 2.**  c = [c \* 3 for c in 'list']  print (c) # ['lll', 'iii', 'sss', 'ttt']  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547299742772/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list.png?attredirects=0)  **Примеры использования генераторов списка.**  **Пример 1.**  Заполнить список квадратами чисел от 0 до 9, используя генератор списка.  Решение:  l = [i\*i for i in range(10)]  **Пример 2.**  Заполнить список числами, где каждое последующее число больше на 2.  l = [(i+1)+i for i in range(10)]  print(l)  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547519534818/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list4.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list4.png?attredirects=0)  Модуль random предоставляет функции для генерации случайных чисел, букв, случайного выбора элементов последовательности. random.randint(A, B) - случайное целое число N, A ≤ N ≤ B. random.random() - случайное число от 0 до 1.  Случайные числа в списке:  10 чисел, сгенерированных случайным образом в диапазоне (10,80) from random import randint  l = [randint(10,80) for x in range(10)]  10 чисел, сгенерированных случайным образом в диапазоне (0,1) l = [random() for i in range(10)]  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547520842756/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list5_res.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list5_res.png?attredirects=0)  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547520783992/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list5.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list5.png?attredirects=0)  **4. Ввод списка (массива) в языке Python.**  Для ввода элементов списка используется цикл for и команда range ():  for i in range(N):     x[i] = int( input() )  **Более простой вариант ввода списка:**  x = [ int(input()) for i in range(N) ]  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547692616791/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_array_prog.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_array_prog.png?attredirects=0)  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547692645732/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_array_res.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_array_res.png?attredirects=0)  Функция int здесь используется для того, чтобы строка, введенная пользователем, преобразовывалась в целые числа.  **5.Вывод списка (массива) в языке Python.**  **Вывод целого списка (массива):**  print (L)  **Поэлементный вывод списка (массива):**  for i in range(N):    print ( L[i], end = " " )  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547518016396/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list_prog2_res.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list_prog2_res.png?attredirects=0)  **2. Методы списков.**   |  |  | | --- | --- | | **Метод** | **Что делает** | | **list.append**(x) | Добавляет элемент в конец списка | | **list.extend**(L) | Расширяет список list, добавляя в конец все элементы списка L | | **list.insert**(i, x) | Вставляет перед i-ым элементом значение x | | **list.remove**(x) | Удаляет первый элемент в списке, имеющий значение x. ValueError, если такого элемента не существует | | **list.pop**([i]) | Удаляет i-ый элемент и возвращает его. Если индекс не указан, удаляется последний элемент | | **list.index**(x, [start [, end]]) | Возвращает положение первого элемента со значением x (при этом поиск ведется от start до end) | | **list.count**(x) | Возвращает количество элементов со значением x | | **list.reverse**() | Разворачивает список | | **list.copy**() | Поверхностная копия списка | | **list.clear**() | Очищает список |   Ниже приведена программа, демонстрирующая методы работы списков.  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547518558801/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list_prog2.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list_prog2.png?attredirects=0)  Пример программы на Python  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547518644911/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list2.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_list2.png?attredirects=0)  Результат выполнения программы  **Вариант 0**  1. Из  массива X  длиной n, среди элементов которого  есть положительные, отрицательные и равные нулю, сформировать новый массив Y, взяв в него только те элементы из X,  которые больше по модулю заданного числа M. Вывести на  экран число M, данный и полученные массивы.  **Решение:**  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547424991714/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_array_prog_var0.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_array_prog_var0.png?attredirects=0)  [https://www.sites.google.com/site/moiboarkin/_/rsrc/1547425095351/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_array_prog_var0_res.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_array_prog_var0_res.png?attredirects=0)  2. В массиве целых чисел все отрицательные элементы заменить на положительные. Вывести исходный массив и полученный.  **Решение:**  **[https://www.sites.google.com/site/moiboarkin/_/rsrc/1547605451968/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_array_var0_2_prog.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_array_var0_2_prog.png?attredirects=0)**  **[https://www.sites.google.com/site/moiboarkin/_/rsrc/1547605554165/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_array_var0_2_res.png](https://sites.google.com/site/moiboarkin/laboratornye-raboty/5-kurs/laboratornye-raboty-po-python/l-r-6/ex_array_var0_2_res.png?attredirects=0)**  **Вариант 1**  1. Дан одномерный массив, состоящий из N целочисленных элементов. Ввести массив с клавиатуры. Найти максимальный элемент. Вывести массив на экран в обратном порядке.  2. В массиве действительных чисел все нулевые элементы заменить на среднее арифметическое всех элементов массива.  **Вариант 2**  1. Дан одномерный массив, состоящий из N целочисленных элементов. Ввести массив с клавиатуры. Найти минимальный элемент. Вывести индекс минимального элемента на экран.  2. Дан массив целых чисел. Переписать все положительные элементы во второй массив, а остальные - в третий.  **Вариант 3**  1. В одномерном числовом массиве D длиной n вычислить сумму элементов с нечетными индексами. Вывести на экран массив D, полученную сумму. 2. Дан одномерный массив из 8 элементов. Заменить все элементы массива меньшие 15 их удвоенными значениями. Вывести на экран монитора преобразованный массив.  **Вариант 4**  1. Дан массив целых чисел. Найти максимальный элемент массива и его порядковый номер. 2. Дан одномерный массив целого типа. Получить другой массив, состоящий только из нечетных чисел исходного массива или сообщить, что таких чисел нет. Полученный массив вывести в порядке убывания элементов.  **Вариант 5**  1. Дан одномерный массив из 10 целых чисел. Вывести пары отрицательных чисел, стоящих рядом. 2. Дан целочисленный массив размера 10. Создать новый массив, удалив все одинаковые элементы, оставив их 1 раз.  **Вариант 6**  1. Дан одномерный массив из 10 целых чисел. Найти максимальный элемент и сравнить с ним остальные элементы. Вывести количество меньших максимального и больших максимального элемента. 2. Одномерный массив из 10-и целых чисел заполнить с клавиатуры, определить сумму тех чисел, которые >5.  **Вариант 7**  1. Дан массив целых чисел. Найти сумму элементов с четными номерами и произведение элементов с нечетными номерами. Вывести сумму и произведение. 2. Переставить в одномерном массиве минимальный элемент и максимальный.  **Вариант 8**  1. Найдите сумму и произведение элементов списка. Результаты вывести на экран. 2. В массиве действительных чисел все нулевые элементы заменить на среднее арифметическое всех элементов массива.  **Вариант 9**  1. Дан одномерный массив, состоящий из N вещественных элементов. Ввести массив с клавиатуры. Найти и вывести минимальный по модулю элемент. Вывести массив на экран в обратном порядке. 2. Даны массивы A и B одинакового размера 10. Вывести исходные массивы. Поменять местами их содержимое и вывести в начале элементы преобразованного массива A, а затем — элементы преобразованного массива B.  **Вариант 10**  1. Определите, есть ли в списке повторяющиеся элементы, если да, то вывести на экран это значение, иначе сообщение об их отсутствии. 2. Дан одномерный массив из 15 элементов. Элементам массива меньше 10 присвоить нулевые значения, а элементам больше 20 присвоить 1. Вывести на экран монитора первоначальный и преобразованный массивы в строчку.  **Вариант 11**  1 Найти наибольший элемент списка, который делиться на 2 без остатка и вывести его на экран. 2. Дан одномерный массив целого типа. Получить другой массив, состоящий только из четных чисел исходного массива, меньше 10, или сообщить, что таких чисел нет. Полученный массив вывести в порядке возрастания элементов.  **Вариант 12**  1. Найти наименьший нечетный элемент списка и вывести его на экран. 2. Даны массивы A и B одинакового размера 10. Поменять местами их содержимое и вывести вначале элементы преобразованного массива A, а затем — элементы преобразованного массива B.  **Вариант 13**  1. Дан одномерный массив целых чисел. Проверить, есть ли в нем одинаковые элементы. Вывести эти элементы и их индексы. 2. Дан одномерный массив из 8 элементов. Заменить все элементы массива меньшие 15 их удвоенными значениями. Вывести на экран монитора преобразованный массив.  **Вариант 14**  1. Найти максимальный элемент численного массива и поменять его местами с минимальным.  2. Программа заполняет одномерный массив из 10 целых чисел числами, считанными с клавиатуры. Определить среднее арифметическое всех чисел массива. Заменить элементы массива большие среднего арифметического на 1.  **Вариант 15**  1. Определите, есть ли в списке повторяющиеся элементы, если да, то вывести на экран эти значения. 2. Дан одномерный массив целого типа. Получить другой массив, состоящий только из нечетных чисел исходного массива или сообщить, что таких чисел нет. Полученный массив вывести в порядке убывания элементов. |