

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (ФГБОУ
ВО «ВВГУ»)

ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И АНАЛИЗА
ДАННЫХ КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И СИСТЕМ

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №5

по дисциплине «Информатика и основы
программирования»

Студент
гр. БИС-25-3
Ассистент
преподавателя

Д. А. Пестерев
М. В. Водяницкий

Задание

Выполнить задания на Python и оформить отчет по стандартам ВВГУ.

Задание 1. Дан список из 10 различных целых чисел. Необходимо найти в нем число 3 и заменить на 30.

Задание 2. Дан список из 5 целых чисел. Необходимо превратить его в список квадратов этих чисел.

Задание 3. Имеется список различных целых чисел. Программа должна найти наибольшее из чисел списка и разделить его на длину списка.

Задание 4. Имеется кортеж из нескольких произвольных элементов. Необходимо этот кортеж отсортировать. Если хотя бы один элемент не является числом, то кортеж остается неизменным.

Задание 5. Имеется словарь товаров в магазине. Необходимо найти товар с минимальной и максимальной ценой.

Задание 6. Имеется список произвольных элементов. Необходимо на основе этого списка создать словарь, где каждый элемент списка будет и ключом, и значением.

Задание 7. Имеется словарь перевода английских слов на русский, где ключ английского слова, значение - русского. Необходимо реализовать программу которая получает на ввод русское слово и результатом выдает перевод на английский.

Задание 8. Реализовать игру Камень-Ножницы-Бумага-Ящерица-Спок. Программа должна запрашивать у пользователя ввод одного из вариантов. Второй вариант случайно генерирует сама программа и возвращает победителя.

Правила игры следующие:

- Ножницы режут бумагу
- Бумага покрывает камень
- Камень давит ящерицу
- Ящерица отравляет Спока
- Спок ломает ножницы
- Ножницы обезглавливают ящерицу
- Ящерица съедает бумагу
- Бумага подставляет Спока
- Спок испаряет камень
- Камень разбивает ножницы

Задание 9. Дан список слов - например:

["яблоко", "груша", "банан", "киви", "апельсин", "ананас"]

Необходимо создать новый словарь, где:

- Ключом будет первая буква слова
- Значением - список всех слов, начинающихся с этой буквы

Задание 10. Дан список кортежей, где каждый кортеж содержит имя студента и его оценки, например:

[("Анна", [5, 4, 5]), ("Иван", [3, 4, 4]), ("Мария", [5, 5, 5])]

Необходимо:

1. Создать словарь, где ключ - имя студента, значение - его средняя оценка
2. Найти студента с наибольшей средней оценкой и вывести его имя и средний балл

Оглавление

1 Выполнение работы	5
1.1 Задание 1.....	5
1.2 Задание 2.....	5
1.3 Задание 3.....	6
1.4 Задание 4.....	6
1.5 Задание 5.....	7
1.6 Задание 6.....	8
1.7 Задание 7.....	9
1.8 Задание 8.....	10
1.9 Задание 9.....	11
1.10 Задание 10.....	11

1 Выполнение работы

1.1 Задание 1

В задании необходимо было написать программу, которая будет проходить по заданному списку, находить в нём цифру 3 и заменять её числом 30. На рисунке 1 представлен код программы.

```

1      # Задание 1
2      nums = [12, 4, 32, 84, 33, 42, 52, 3, 6, 5]
3      a = -1
4      for i in nums:
5          a += 1
6          if i == 3:
7              nums[a] = 30
8      print(nums)

```

Рисунок 1 – Листинг программы для задания 1

Пояснение работы программы:

- 1) Задаём список в переменную `nums`
- 2) Вводим переменную `a` для подсчёта номера элемента.
- 3) Далее с помощью цикла `for` перебираем каждый элемент в списке
- 4) С помощью условия `if` проверяем каждый элемент на равенство трём, после того как находим такой элемент заменяем его на 30 и выводим на экран обновлённый список.

1.2 Задание 2

В данном задании необходимо было написать программу, которая проходит по заданному списку и заменяет все числа списка на квадраты этих чисел. На рисунке 2 представлен код программы.

```

1      # Задание 2
2      nums1 = [12, 4, 32, 84, 33]
3      a = -1
4      for i in nums1:
5          a += 1
6          nums1[a] = i * i
7
8      print(nums1)

```

Рисунок 2 – Листинг программы для задания 2

Пояснение работы программы:

- 1) Задаём произвольный список элементов.
- 2) Вводим переменную для подсчёта номера элемента.
- 3) С помощью цикла перебираем список и каждый элемент заменяем на его квадрат посредством умножения каждого элемента на себя же.
- 4) На экран выводится обновлённый список квадратов изначальных чисел.

1.3 Задание 3

В данном задании необходимо было написать программу которая будет находить наибольшее из чисел списка и делить его на количество элементов списка. На рисунке 3 представлен код программы.

```

1      # Задание 3
2      nums2 = [1, 15, 3]
3      a = 0
4      b = 0
5      for i in nums2:
6          a += 1
7          if i > b:
8              b = i
9
10     print(b / a)

```

Рисунок 3 – Листинг программы для задания 3

Пояснение работы программы:

- 1) Задаём произвольный список элементов.
- 2) Создаётся переменная *a* для подсчёта количества элементов, а также переменная *b* для записи наибольшего числа.
- 3) С помощью цикла for перебираем список и проверяем больше ли проверяемый элемент чем *b*, если да, то присваеваем значение элемента.
- 4) На экран выводится наибольшее число делённое на количество элементов.

1.4 Задание 4

В этом задании необходимо было написать программу, которая будет сортировать кортеж . На рисунке 4 представлен код программы.

```
1      # Задание 4
2      data = (3, 'f', 2)
3
4      if all(isinstance(x, (int, float)) for x in data):
5          sorted_data = tuple(sorted(data))
6      else:
7          sorted_data = data
8
9      print(sorted_data)
```

Рисунок 4 – Листинг программы для задания 4

Пояснение работы программы:

- 1) При помощи функции `input()` запрашивается число и записывается в переменную `number`.
- 2) Далее мы создаём список `mas` проходя по каждому элементу строки `number` и записывая его в целочисленном виде.
- 3) Затем мы с помощью условия `if` проходимся по правилам деления на 6, а именно проверяем чётная ли последняя цифра числа, а также делится ли сумма всех цифр на 3.
- 4) По завершению на экран выводится сообщение о том, делится ли введённое число на 6 или нет.

1.5 Задание 5

В данном задании необходимо было написать программу, которая будет проходить по словарю товаров магазина и находить в нём товар с наименьшей и наибольшей ценой. На рисунке 5 представлен код программы.

```

1      # Задание 5
2      d = {"Молоко": 100, "сыр": 350, "колбаса": 300}
3
4      min_cost = 0
5      min_name = None
6      max_cost = 0
7      max_name = None
8
9      for key, value in d.items():
10         if min_cost == 0:
11             min_cost = value
12             min_name = key
13         if value < min_cost:
14             min_cost = value
15             min_name = key
16
17         for key, value in d.items():
18             if value > max_cost:
19                 max_cost = value
20                 max_name = key
21
22     print(f'Продукт с наименьшей ценой: {min_name}')
23     print(f'Продукт с наибольшей ценой: {max_name}')

```

Рисунок 5 – Листинг программы для задания 5

Пояснение работы программы:

- 1) Сначала задаём произвольный словарь товаров.
- 2) Далее создаём переменные для продуктов с минимальной и максимальной стоимостью, а также отдельные переменные для их цены.
- 3) Для начала с помощью цикла for определим продукт с минимальной ценой: для этого перебираем ключи и значения словаря, задаём для минимальной цены значение value, а для имени товара присваиваем значение ключа.
- 4) Затем следующим циклом находим товар с максимальной ценой сравнивая проверяемое значение с максимальным.
- 5) На экран выводится сначала продукт с наименьшей ценой, а затем с наибольшей ценой.

1.6 Задание 6

В данном задании необходимо было написать программу, которая будет на основе данного списка делать словарь, где каждый элемент списка будет и ключом, и значением. На рисунке 6 представлен код программы.

```

1      # Задание 6
2      lst = ['молоко', 'колбаса', 'сыр', 1]
3      lst_dict = {}
4
5      for i in lst:
6          lst_dict[i] = i
7
8      print(lst_dict)

```

Рисунок 6 – Листинг программы для задания 6

Пояснение работы программы:

- 1) Задаём произвольный список lst.
- 2) Далее создаём пустой словарь для последующего заполнения.
- 3) Циклом перебераем все элементы в нашем списке, и каждый раз добавляем в наш словарь ключ и значение равное перебираемому элементу.
- 4) На экран выводится заполненный словарь, где каждый элемент и ключ и значение.

1.7 Задание 7

В данном задании необходимо было написать программу, которая получает на ввод русское слово и результатом выдает перевод на английский путём перебора словаря. На рисунке 7 представлен код программы.

```

1      # Задание 7
2      translate = {'молоко': 'milk', 'сыр': 'cheese', 'торт': 'cake'}
3
4      word = input('Введите слово: ')
5      print(translate.get(word, 'Такого слова нет!'))

```

Рисунок 7 – Листинг программы для задания 7

Пояснение работы программы:

- 1) Задаём словарь где ключём является русское слово, а значением - перевод этого слова на английский.
- 2) С помощью функции input запрашивается слово на русском.
- 3) Далее с помощью метода словаря get() берётся значение по ключу(word), затем это значение выводится на экран, если ключа нет – выводит второй аргумент “такого слова нет”.

1.8 Задание 8

В задании необходимо было написать программу, которая будет реализовывать игру, программа будет запрашивать ввод одного варианта у пользователя с консоли, а второй вариант сама программа будет генерировать рандомно. На рисунке 8 представлен код программы.

```

1   # Задание 8
2   import random
3   game = [
4       ('ножницы', 'бумага'), ('ножницы', 'ящерица'),
5       ('бумага', 'камень'), ('бумага', 'спок'),
6       ('камень', 'ящерица'), ('камень', 'ножницы'),
7       ('ящерица', 'спок'), ('ящерица', 'бумага'),
8       ('спок', 'ножницы'), ('спок', 'камень')
9   ]
10  g1 = input('g1 выбор хода: ')
11  g2 = random.choice(random.choice(game))
12  winner = None
13
14  for first, second in game:
15      if first == g1:
16          if second == g2:
17              winner = g1
18      else:
19          if first == g2:
20              if second == g1:
21                  winner = g2
22
23  print(f'g2: {g2}')
24  print(f'победитель {winner}')

```

Рисунок 8 – Листинг программы для задания 8

Пояснение работы программы:

- 1) Для начала импортируем модуль random для случайного выбора хода компьютера.
- 2) Далее создаём список кортежей со всеми возможными вариациями победы, где первый выбор доминирует над вторым.
- 3) С помощью функции input запрашиваем в консоль выбор хода, далее с помощью функции random.choice выбираем случайную пару, затем повторяя функцию выбираем уже случайный элемент из выбранной пары. Также задаём переменную winner в которую будем записывать победителя.

4) Создаём цикл for в котором перебираем элементы кортежей, если первый элемент кортежа равен выбору хода с консоли, а второй равен случайному ходу компьютера, тогда победитель игрок, в противоположном случае – компьютер.

5) На экран выводится выбор компьютера, а затем победитель

1.9 Задание 9

В этом задании необходимо было написать программу, которая будет на основе данного списка создавать словарь, в котором ключём будет первая буква слова, а значением список всех слов начинающихся с этой буквы. На рисунке 9 представлен код программы.

```

1      # Задание 9
2      shop = ['молоко', 'сыр', 'колбаса', 'пирог', 'каша', 'яблоки', 'сырники']
3      shop_dict = {}
4
5      for i in shop:
6          key = i[0]
7          if key not in shop_dict:
8              shop_dict[key] = []
9              shop_dict[key].append(i)
10
11     print(shop_dict)

```

Рисунок 9 – Листинг программы для задания 9

Пояснение работы программы:

- 1) Задаём произвольный список продуктов, также создаём пустой словарь.
- 2) С помощью цикла for начинаем перебирать элементы нашего списка.
- 3) Для этого сначала делаем ключём первую букву элемента, далее проверяем, что если ключа нет в словаре и добавляем его, а также создаём для этого ключа пустой список.
- 4) Далее добавляем перебираемый элемент в наш список, таким способом в список будут попадать все элементы первая буква которых является ключём.
- 5) На экран выводится заполненный словарь.

1.10 Задание 10

В задании необходимо было написать программу, которая будет из списка кортежей создавать словарь, где ключ - имя студента, значение - его средняя оценка, а также находить студента с наибольшей средней оценкой и выводить его имя и средний балл. На рисунке 10 представлен код программы.

```

1  # Задание 10
2  students = [("Пётр", [5, 4, 3]), ("Даниил", [3, 4, 4]), ("Александр", [4, 5, 5])]
3  best = {}
4  mid = 0
5  a = 0
6
7  for key, value in students:
8      for i in value:
9          a += 1
10         mid += i
11     mid /= a
12     best[key] = round(mid, 1)
13     a = 0
14     mid = 0
15
16 best_student = max(best, key=best.get)
17 print(f'{best_student} имеет наивысший средний балл: {best[best_student]}')

```

Рисунок 10 – Листинг программы для задания 10

Пояснение работы программы:

- 1) С помощью функции `input()` у пользователя запрашивается число, которое сохраняется в переменной `number`.
- 2) Создаётся переменная `d`, для подсчёта количества делителей.
- 3) Далее делаем проверку на то, что в консоль введено не целое число, если условие соблюдается выдаём на экран соответствующую ошибку. метод `.isdigit()` возвращает `True`, если строка содержит только числа.
- 4) Иначе продолжаем проверку дальше, проверяем, что в консоль введено число не больше единицы, если условие соблюдается выдаём на экран соответствующую ошибку.
- 5) Иначе вычисляем количество делителей числа.
- 6) На экран выводится сообщение, соответствующее количеству делителей, если количество делителей равно двум, то число простое, иначе составное.