

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Кафедра экономической информатики

Лабораторная работа №1

«Основы программирования на Python»

Выполнил:

студент гр. 414302
К. С. Черненко

Проверил:

Мацокин М П.

Минск 2025

Цель работы: закрепление на практике теоретических знаний по основам программирования на Python путем решения комплекса задач.

Вариант 29

Задание 1.

Определить, сколько в числе четных цифр, а сколько нечетных.

Число вводится с клавиатуры.

Листинг 1

```
even = 0
odd = 0

number = int(input("Enter a number: "))
number = abs(number)

if number == 0:
    even = 1
else:
    while number > 0:
        digit = number % 10
        if digit % 2 == 0:
            even += 1
        else:
            odd += 1
        number //= 10

print("Even:", even)
print("Odd:", odd)
```

Результата работы программы:

```
/home/demzez/BSUIR/SPL/Lab1/.venv/bin/python /home/demzez/BSUIR/SPL/Lab1/Lab1.1.py
Enter a number: 123
Even: 1
Odd: 2

Process finished with exit code 0
```

Задание 2.

Посчитать, сколько пар (стоят рядом) верхнего и нижнего регистра находится в введенном с клавиатуры слове. (Пример HjkLM- 1 пара нижнего, 1 пара верхнего), а также сколько всего букв в слове.

Листинг 2

```
upperPairs = 0
lowerPairs = 0

word = input("Enter a word: ")
for i in range(len(word)-1):
    letter1 = word[i]
    letter2 = word[i+1]

    if letter1.isupper() and letter2.isupper():
        upperPairs += 1
    elif letter1.islower() and letter2.islower():
        lowerPairs += 1
    else:
        continue

print(f"Count of letters: {len(word)}")
print(f"Upper Pairs: {upperPairs}")
print(f"Lower Pairs: {lowerPairs}")
```

Результата работы программы:

```
/home/demzez/BSUIR/SPL/Lab1/.venv/bin/python /home/demzez/BSUIR/SPL/Lab1/Lab1.2.py
Enter a word: WThDjj
Count of letters: 6
Upper Pairs: 1
Lower Pairs: 1

Process finished with exit code 0
```

Задание 3.

Найдите сумму положительных элементов списка.

Найдите сумму элементов списка между двумя первыми нулями. Если двух нулей нет в списке, то выведите ноль.

Листинг 3

```
list1 = [1, -2, 3, 0, 4, -4, -1, 3, -2, 4, 0, -3, 2, 6]
sumPositiveElements = 0
sumZeroElements = 0
isOk = False
zeroCount = 0

for element in list1:
    if element > 0:
        sumPositiveElements += element
```

```

    if element == 0:
        zeroCount += 1
        if zeroCount <= 2:
            isOk = not isOk
            continue
        if isOk:
            sumZeroElements += element

print(f'Positive: {sumPositiveElements}')
if zeroCount <= 1:
    print(f'No elements between zero numbers. {0}')
else:
    print(f'Zero: {sumZeroElements}')

```

Результата работы программы:

```

/home/demzez/BSUIR/SPL/Lab1/.venv/bin/python /home/demzez/BSUIR/SPL/Lab1/Lab1.3.py
Positive: 23
Zero: 4

Process finished with exit code 0

```

Задание 4.

Анаграммами называются слова, образованные путем взаимной перестановки букв. В английском языке, например, анаграммами являются слова «live» и «evil», а в русском – «выбор» и «обрыв». Напишите программу, которая будет запрашивать у пользователя два слова, определять, являются ли они анаграммами, и выводить на экран ответ.

Листинг 4

```

firstWord = input("Enter first word: ").lower()
secondWord = input("Enter second word: ").lower()

charsWord1 = list(firstWord)
charsWord2 = list(secondWord)

n1 = len(charsWord1)
n2 = len(charsWord2)

for i in range(max(n1, n2)):

```

```

if i < n1:
    for j in range(0, n1 - i - 1):
        if charsWord1[j] > charsWord1[j + 1]:
            charsWord1[j], charsWord1[j + 1] = charsWord1[j + 1], charsWord1[j]

if i < n2:
    for j in range(0, n2 - i - 1):
        if charsWord2[j] > charsWord2[j + 1]:
            charsWord2[j], charsWord2[j + 1] = charsWord2[j + 1], charsWord2[j]

if charsWord1 == charsWord2:
    print("Anagrams!")
else:
    print("Not anagrams!")

```

Результата работы программы:

```

/home/demzez/BSUIR/SPL/Lab1/.venv/bin/python /home/demzez/BSUIR/SPL/Lab1/Lab1.4.py
Enter first word: evil
Enter second word: live
Anagrams!

Process finished with exit code 0

```

Задание 5.

Реализуйте программу «Магазин автозапчастей», которая будет включать в себя шесть пунктов меню. У вас есть словарь, где ключ – название продукции. Значение – список, который содержит состав продукции, цену и кол-во (шт), которое есть в магазине.

1. Просмотр описания: название – описание
2. Просмотр цены: название – цена.
3. Просмотр количества: название – количество.
4. Вся информацию.
5. Покупка

В пункте «Покупка» необходимо совершить покупку, с клавиатуры вводите название продукции и его кол-во, n – выход из программы. Посчитать цену выбранных товаров и сколько товаров осталось в изначальном списке. На выходе должен быть чек.

6. До свидания

Листинг 5

```
products = {
    "Масло моторное": ["Синтетическое масло 5W-30", 2500, 15],
    "Воздушный фильтр": ["Бумажный фильтр", 1200, 8],
    "Тормозные колодки": ["Керамические колодки", 4500, 5],
    "Аккумулятор": ["Свинцово-кислотный 60Ah", 8000, 3],
    "Свечи зажигания": ["Иридиевые свечи", 2000, 20]
}

def view_descriptions():
    print("\n--- ОПИСАНИЯ ТОВАРОВ ---")
    for product, info in products.items():
        print(f'{product} - {info[0]}')

def view_prices():
    print("\n--- ЦЕНЫ ТОВАРОВ ---")
    for product, info in products.items():
        print(f'{product} - {info[1]} руб.')

def view_quantities():
    print("\n--- КОЛИЧЕСТВО ТОВАРОВ ---")
    for product, info in products.items():
        print(f'{product} - {info[2]} шт.')

def view_all_info():
    print("\n--- ВСЯ ИНФОРМАЦИЯ ---")
    for product, info in products.items():
        print(f'{product}: {info[0]}, Цена: {info[1]} руб., Количество: {info[2]} шт.')

def make_purchase():
    print("\n--- ПОКУПКА ---")
    print("Введите название товара и количество через пробел")

    total_price = 0
    receipt = []

    while True:
        # Проверки
        user_input = input("Товар и количество (или 'н' для выхода): ").strip()
        if user_input.lower() == 'н':
            break

        parts = user_input.split()
        if len(parts) < 2:
```

```

    print("Ошибка: введите название и количество через пробел")
    continue

    product_name = ''.join(parts[:-1]) # объединение нескольких элементов кроме
последнего в одну строку
    if parts[-1].isdigit():
        count = int(parts[-1])
    else:
        print("Ошибка: количество должно быть числом")
        continue

    if product_name not in products:
        print(f"Ошибка: товар '{product_name}' не найден")
        continue

    product_info = products[product_name]
    if count > product_info[2]:
        print(f"Ошибка: недостаточно товара. В наличии: {product_info[2]} шт.")
        continue

    if count <= 0:
        print("Ошибка: количество должно быть положительным")
        continue

    # Покупка
    price = product_info[1] * count
    total_price += price
    product_info[2] -= count

    # Добавляем в чек
    receipt.append((product_name, count, price))
    print(f"Добавлено: {product_name} x{count} = {price} руб.")

    # Печатаем чек
    if receipt:
        print("\n=== ВАШ ЧЕК ===")
        print("Товар\tКоличество\tСтоимость")
        print("-" * 40)
        for item in receipt:
            print(f"{item[0]}\t{item[1]} шт.\t{item[2]} руб.")
            print("-" * 40)
        print(f"ИТОГО: {total_price} руб.")
        print("Спасибо за покупку!")
    else:
        print("Покупка отменена")

while True:

```

```

print("\n=== МАГАЗИН АВТОЗАПЧАСТЕЙ ===")
print("1. Просмотр описания")
print("2. Просмотр цены")
print("3. Просмотр количества")
print("4. Вся информация")
print("5. Покупка")
print("6. До свидания")

choice = input("Выберите пункт меню (1-6): ")

if choice == "1":
    view_descriptions()
elif choice == "2":
    view_prices()
elif choice == "3":
    view_quantities()
elif choice == "4":
    view_all_info()
elif choice == "5":
    make_purchase()
elif choice == "6":
    print("До свидания! Приходите еще!")
    break
else:
    print("Неверный выбор. Попробуйте снова.")

```

Результата работы программы:

```

Масло моторное: Синтетическое масло 5W-30, Цена: 2500 руб., Количество: 15 шт.
Воздушный фильтр: Бумажный фильтр, Цена: 1200 руб., Количество: 8 шт.
Тормозные колодки: Керамические колодки, Цена: 4500 руб., Количество: 5 шт.
Аккумулятор: Свинцово-кислотный 60Ah, Цена: 8000 руб., Количество: 3 шт.
Свечи зажигания: Иридиевые свечи, Цена: 2000 руб., Количество: 20 шт.

=== МАГАЗИН АВТОЗАПЧАСТЕЙ ===
1. Просмотр описания
2. Просмотр цены
3. Просмотр количества
4. Вся информация
5. Покупка
6. До свидания
Выберите пункт меню (1-6): 5

--- ПОКУПКА ---
Введите название товара и количество через пробел
Товар и количество (или 'n' для выхода): Воздушный фильтр 4
Добавлено: Воздушный фильтр x4 = 4800 руб.
Товар и количество (или 'n' для выхода): Масло моторное 10
Добавлено: Масло моторное x10 = 25000 руб.
Товар и количество (или 'n' для выхода): n

=== ВАШ ЧЕК ===
Товар      Количество  Стоимость
-----
Воздушный фильтр   4 шт.      4800 руб.
Масло моторное    10 шт.     25000 руб.
-----
ИТОГО: 29800 руб.
Спасибо за покупку!

```


Задание 6.

Определить, сколько в числе четных цифр, а сколько нечетных.
Число вводится с клавиатуры.

Листинг 6

```
tupleList = (1, 2, 3, 4, 5)
minEvenElement = tupleList[0]
isOk = True

for element in tupleList:
    if element % 2 == 0 and isOk == True:
        minEvenElement = element
        isOk = False
    if element < minEvenElement and element % 2 == 0:
        minEvenElement = element

print(f"Минимальный четный элемент картежа: {minEvenElement}")
```

Результата работы программы:

```
/home/demzez/BSUIR/SPL/Lab1/.venv/bin/python /home/demzez/BSUIR/SPL/Lab1/Lab1.6.py
Минимальный четный элемент картежа: 2

Process finished with exit code 0
```

Вывод:

В результате выполнения лабораторной работы были успешно закреплены практические навыки программирования на Python, освоены основные структуры данных и алгоритмы их обработки.