

Politechnika Warszawska
Wydział Elektroniki i Technik Informacyjnych

PROS PYTHON LAB3

Autorka:
NATALIA ŚLEPOWROŃSKA 318847

Warszawa 2024Z

Spis treści

1. Wstęp	2
2. Opis działania programu	2
2.1. Funkcja inicjująca	2
2.2. Funkcja wczytująca dane	2
2.3. Funkcje przetwarzające dane	3
2.4. Funkcja wyświetlająca dane	4
3. Schemat blokowy	5
4. Wynik działania programu	6

1. Wstęp

W ramach pierwszego laboratorium dotyczącego języka python, opracowałam program do prostej analizy finansowej. Program ten pobiera od użytkownika dochody dla danego miesiąca, a następnie dla zadanaego okresu przez użytkownika, oblicza on średnią oraz medianę zarobków.

2. Opis działania programu

Na początku działania programu importuję bibliotekę **json** oraz deklaruję zmienną **months** jest to tablica, zmienna globalna.

Listing 1: Deklaracja zmiennych i bibliotek na początku programu

```
1 import json
2
3 months = []
```

2.1. Funkcja inicjująca

W funkcji **initialize** odwołuję się do zmiennej globalnej. Tablicę **months** wypełniam nazwami miesięcy. Funkcja ta nie posiada argumentów oraz nie zwraca wartości.

Listing 2: Funkcja initialize

```
1 def initialize():
2     global months
3     months = ["styczen", "luty", "marzec", "kwiecien", "maj", "czerwiec",
4               "lipiec", "sierpnien", "wrzesien", "pazdziernik", "listopad", "grudzien"]
```

2.2. Funkcja wczytująca dane

W funkcji **input_incomes** tworzę odwołania do zmiennej globalnej i tworzę słownik **incomes**. Następnie pobieram od użytkownika liczbę miesięcy do analizy finansowej. Program sprawdza, czy liczba znajduje się w zakresie od 0 do 12. Jeżeli liczba znajduje się poza zakresem, program prosi o ponowne podanie liczby.

Listing 3: Funkcja input_incomes, część I

```
1 def input_incomes():
2     incomes = {}
3     global months
4
5     months_num = int(input("Podaj liczbe miesiecy (od początku roku)
6                           dla ktorych chcesz wykonac analize finansowa: "))
7
8     if months_num < 0 or months_num > 12:
9         months_num = int(input("Podaj poprawna liczbe miesiecy: "))
10    ...
```

Następnie w pętli for pobieram dla każdego miesiąca dochód i umieszczam go w słowniku **incomes**, gdzie kluczami są nazwy miesięcy. Program sprawdza, czy dochód jest liczbą dodatnią. Jeżeli nie, prosi o ponowne podanie dochodu.

Listing 4: Funkcja input_incomes, część II

```
1    ...
2    for i in range(months_num):
3        user_input = input("Podaj dochod w zlotowkach dla miesiaca {}: "
4                           .format(months[i]))
5        income = float(user_input)
```

```

6
7     if 0 > income:
8         user_input = input("Dochod musi byc rowny badz wiekszy niz 0 zl ,
9                             podaj dochod: ")
10        income = float(user_input)
11        incomes[months[i]] = income
12    else:
13        incomes[months[i]] = income
14    ...

```

Listing 5: Funkcja input_incomes, część III

```

1    ...
2    try:
3        f = open("data.json", "w")
4    except:
5        print("Nie udalo sie zapisac danych do pliku")
6    else:
7        json.dump(incomes, f)
8        f.close()

```

Dane są zapisywane do pliku json. Korzystam z **try** i **except** do obsługi wyjątków w trakcie otwierania pliku. Zapis do pliku jest wykonywany poza operatorami obsługującymi wyjątki.

Funkcja ta nie zawiera argumentów.

2.3. Funkcje przetwarzające dane

Do analizy finansowej wykorzystuję funkcje obliczającą średnią oraz funkcję obliczającą medianę. Funkcje te nie odwołują się do zmiennych globalnych.

Dane są odczytywane z pliku json. Korzystam z **try** i **except** do obsługi wyjątków w trakcie otwierania pliku. Odczyt z pliku oraz reszta obliczeń jest wykonywany poza operatorami obsługującymi wyjątki.

Listing 6: Funkcja average_income

```

1 def average_income():
2     try:
3         f = open("data.json", "r")
4     except:
5         print("Brak danych")
6         return "Brak danych"
7     else:
8         dict = json.load(f)
9         f.close()
10
11        if not dict:
12            print("Brak danych")
13            return "Brak danych"
14
15        l = len(dict)
16        su = sum(dict.values())
17        average = su / l
18
19        return average

```

Listing 7: Funkcja median_income

```

1 def median_income():
2     try:
3         f = open("data.json", "r")

```

```

4     except:
5         print("Brak danych")
6         return "Brak danych"
7     else:
8         dict = json.load(f)
9         f.close()
10
11         if not dict:
12             print("Brak danych")
13             return "Brak danych"
14
15         sorted_data = sorted(dict.values())
16         len_data = len(dict)
17
18         if len_data % 2 == 1:
19             median = sorted_data[len_data // 2]
20         else:
21             mid1 = sorted_data[(len_data // 2) - 1]
22             mid2 = sorted_data[len_data // 2]
23             median = (mid1 + mid2) / 2
24
25         return median

```

Funkcje te sprawdzają, czy zmienna `dict` zawiera dane. Są one wyłowywane wewnątrz funkcji **`display_results`**.

2.4. Funkcja wyświetlająca dane

Do wyświetlenia analizy finansowej wykorzystuję funkcję **`display_results`**.

Listing 8: Funkcja `display_results`

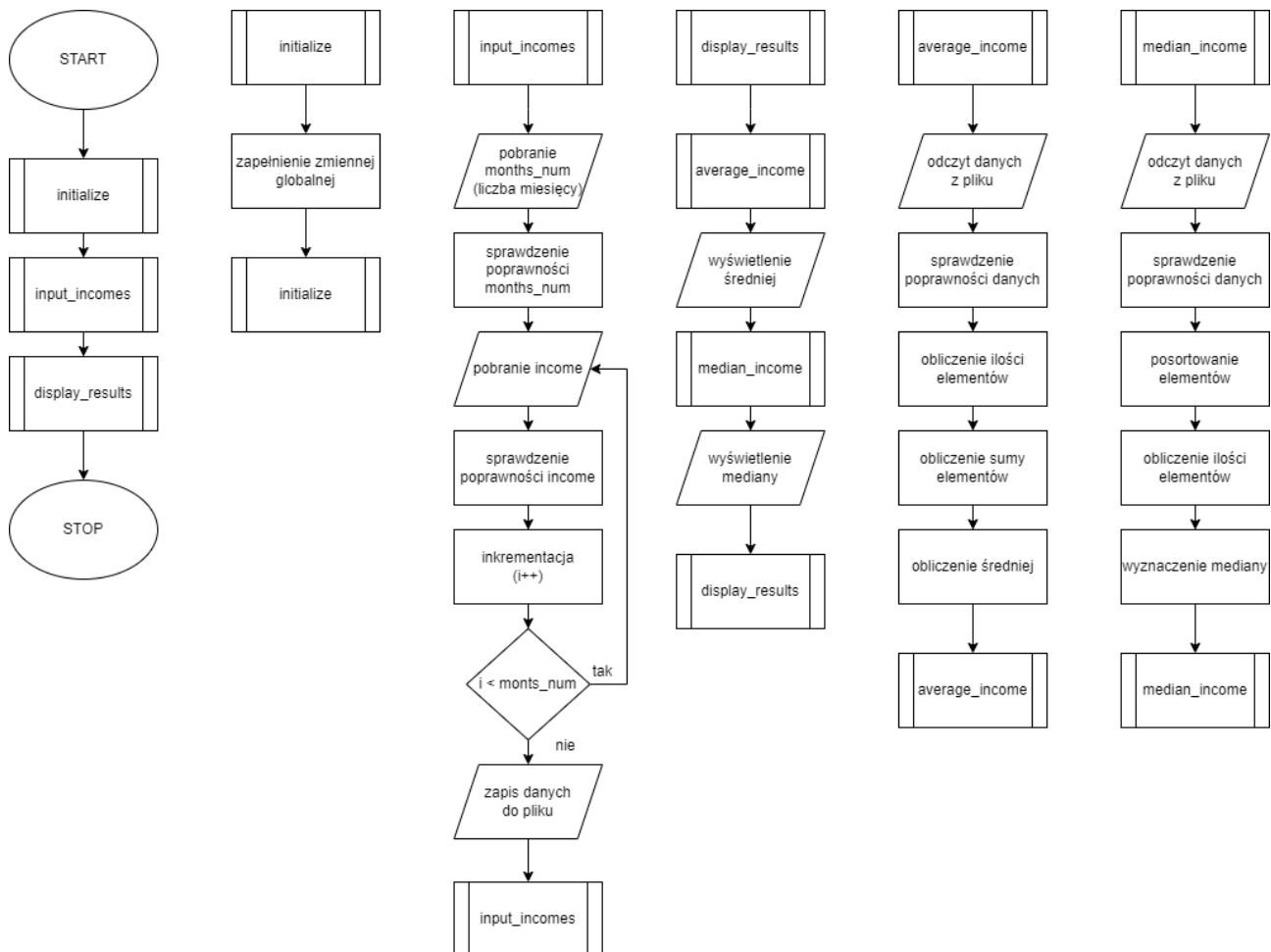
```

1 def display_results():
2     average = average_income()
3     median = median_income()
4     print("Srednia zarobkow dla podanego okresu: {}".format(average))
5     print("Mediana zarobkow dla podanego okresu: {}".format(median))

```

3. Schemat blokowy

Poniżej przedstawiam schemat blokowy mojego programu:



Rysunek 1: Schemat

4. Wynik działania programu

Poniżej przedstawiam wynik działania mojego programu. Przetestowałam różne scenariusze wprowadzenia danych wejściowych.

```
root@natalia:~# python3 lab3.py
Podaj liczbe miesiecy (od poczatku roku) dla ktorych chcesz wykonac analize finansowa: -2
Podaj poprawna liczbe miesiecy: 0
Brak danych
Brak danych
Srednia zarobkow dla podanego okresu: Brak danych
Mediana zarobkow dla podanego okresu: Brak danych
```

Rysunek 2: Niepoprawna a następnie zerowa liczba podanych miesięcy

```
root@natalia:~# python3 lab3.py
Podaj liczbe miesiecy (od poczatku roku) dla ktorych chcesz wykonac analize finansowa: 3
Podaj dochod w zlotowkach dla miesiaca styczen: -3
Dochod musi byc rowny badz wiekszy niz 0 zl, podaj dochod: 1000
Podaj dochod w zlotowkach dla miesiaca luty: 2000
Podaj dochod w zlotowkach dla miesiaca marzec: 5000
Srednia zarobkow dla podanego okresu: 2666.6666666666665
Mediana zarobkow dla podanego okresu: 2000.0
```

Rysunek 3: Niepoprawny dochód a następnie poprawna analiza

Program działa zgodnie z moimi założeniami.