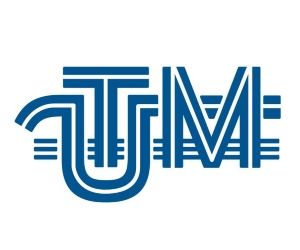
Ministerul Educației și Cercetării al Republicii Moldova

Universitatea Tehnică a Moldovei

Facultatea Calculatoare, Informatică și Microelectronică

Departamentul Ingineria Software și Automatică

**Raport**

la disciplina **”Programare Declarativa”**

Tema: **Introducere în Python**

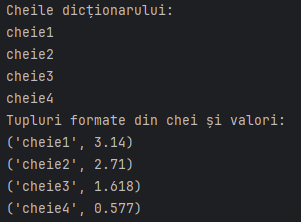
Efectuat de: studentul/studenta gr.**TI-216 Untila Victor**

Verificat de: asis.univ. **Viorel Rusu**

Chișinău-2023

**Exercitiul 1.**

Definiți un dicționar ce folosește șiruri de caractere pe post de chei și elemente float pe post de valori. Afișați doar cheile dicționarului și mai apoi tupluri formate din chei și valori.

********

*Figura 1*. Cheile dictionarului si tupluri din acestea

**Exercitiul 2.**

Generați un număr aleator între 0 și 10000, ce reprezintă un număr de secunde. Calculați reprezentarea numărului de secunde în ore, minute și secunde și afișați rezultatul formatat sub forma hh:mm:ss. Alternativ, folos iți modulul datetime.

****

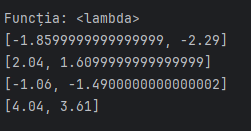
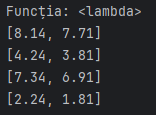


*Figura 2*. Reprezentarea secundelor in format hh:mm:ss

**Exercitiul 3.**

Definiți două obiecte de tip float și calculați suma, diferența lor utilizand funcții lamda(). Apelați toate funcțiile din listă pe rând și cu alte tipuri de date, folosiți funcția map().

****



*Figura 3*. Rezultatele functiilor lambda pentru suma si diferenta

**Exercitiul 4.**

Definiți o listă de valori întregi și afișați doar valorile distincte din aceasta (folosiți funcția filter()).

****



*Figura 4*. Valorile distincte din lista

**Exercitiul 5.**

Definiți o funcție ce returnează numărul de apariții ale unui caracter într-un string (folosiți funcția reduce()).

****



*Figura 5*. Numarul de aparitii a caracterului „e”

**Exercitiul 6.**

Definiți o funcție ce concatenează oricâte stringuri sunt date la intrarea sa.

****

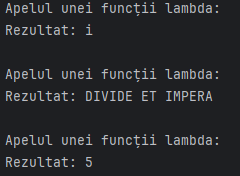


*Figura 6*. Rezultatul functiei de concatenare

**Exercitiul 7.**

Definiți o listă de funcții lambda ce returnează: al doilea caracter dintr-un string; stringul cu litere majuscule; poziția pe care se găsește un anumit caracter dat la intrare. Apelați toate funcțiile din listă pe rând.

****

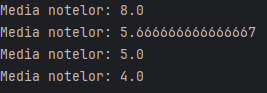
****

*Figura 7*. Rezultatul sarcinii nr.7

**Exercitiul 8.**

Definiți o funcție ce calculează media a trei note sprecificate la intrare. Dacă la apel nu se trimit toate notele, se vor folosi valori implicite egale cu 4. Apelați funcția cu diferite combinații de argumente poziționale și keyword

****



*Figura 8*. Media notelor conform sarcina nr.8

**Exercitiul 9.**

Definiți o funcție recursivă ce afișează suma primelor N numere naturale

****



*Figura 9.* Suma primelor 20 numere naturale

**Concluzie**

In lucrarea curenta am studiat conceptele esențiale legate de funcțiile de ordin înalt, cum ar fi `map()`, `filter()`, `reduce()`, și funcțiile anonime `lambda()`. `map()` aplică o funcție la fiecare element al unei secvențe, `filter()` filtrează elemente în funcție de o condiție, `reduce()` combină elemente succesive într-o singură valoare. Funcțiile `lambda()` permit definirea rapidă a funcțiilor anonime. Aceste concepte sunt puternice și flexibile în programarea Python, facilitând manipularea și transformarea datelor.