

Programare Procedurală

– Laborator 2 –



OUTPUT

- (a) formatare cu ‘%’
- (b) metoda format: Python >= 2.6, Python >= 3.0
- (c) f-string-uri: Python >= 3.6

În rezolvarea exercițiilor următoare exersați toate cele trei forme de afișare.

I. Instrucțiunea condițională

1. Se citesc de la tastatură, de pe câte o linie, trei numere naturale z, l, a , reprezentând ziua, luna și anul unei date calendaristice. Să se afișeze data zilei următoare, în formatul $zi.luna.an$.
Reamintim că un an este bisect dacă:
 - este divizibil cu 4 și nu este divizibil cu 100
 - sau
 - este divizibil cu 400.

II. Instrucțiuni repetitive – algoritmi elementari

Putem folosi și **import math** și **math.sqrt(x)**.

1. Fie x număr natural, citit. Să se verifice dacă x este:
 - a. număr perfect (egal cu suma divizorilor lui, mai puțin numărul însuși; ex.: 6, 28, 496, 8128);
 - b. palindrom (este egal cu răsturnatul său).
2. Fie n număr natural, citit. Să se afișeze:
 - a. $n!$ (Testați cât de mare poate fi valoarea lui n pentru care se calculează $n!$ Ce observați?);
 - b. toate numerele prime mai mici decât n ;
 - c. primele n numere naturale prime;
 - d. primele n pătrate perfecte;
 - e. toate pătratele perfecte mai mici decât n .
3. Se citește un sir de cel puțin 3 valori întregi până la întâlnirea lui 0. Să se afișeze:
 - a. valoarea maximă din sir;
 - b. media aritmetică a valorilor pozitive;
 - c. numărul valorilor de tip “deal” / „vale” (ce au la dreapta și la stânga valori mai mici / mai mari; două valori alăturate în sir vor fi distințe; pentru primul element din sir se va lua în considerare doar valoarea din dreapta, iar pentru ultimul element numai valoarea din stânga); se va ignora valoarea 0 ce încheie sirul.
Exemplu: pentru sirul 3 9 10 -1 3 4 5 4 2 0 avem outputul 2, 3 (valori „deal” 10, 5, valori „vale” 3, -1, 2).
4. Să se genereze toate submulțimile mulțimii $A = \{1, 2, \dots, n\}$, unde numărul natural nenul n se citește de la tastatură (*fără backtracking* – folosind scrierea binară a unui număr și operații pe biți).
Exemplu: pentru $n = 3$ outputul va fi {}, {1}, {2}, {3}, {1, 2}, {1, 3}, {2, 3}, {1, 2, 3}, cu submulțimile afișate în orice ordine.