

SEMINAR 4

Funcții

1.

a) Scrieți o funcție care returnează o matrice triunghiulară de dimensiune n , având forma următoare:

- prima coloană conține numerele $1, 2, 3, \dots, n$;
- ultima linie conține numerele $n, n - 1, \dots, 2, 1$;
- restul elementelor aflate în triunghiul de sub diagonală principală se calculează ca sumă a elementelor vecine de la vest, sud și sud-vest.

b)

Scrieți o funcție care afișează o matrice triunghiulară, memorată sub forma unei liste de liste, în formă matriceală, cu elementele de pe fiecare coloană aliniate la dreapta.

Exemplu: pentru $n = 4$ matricea cerută este $M = [[1], [2, 15], [3, 10, 15], [4, 3, 2, 1]]$ sau, afișată în formă matriceală:

$$M = \begin{pmatrix} 1 & & & \\ 2 & 15 & & \\ 3 & 10 & 15 & \\ 4 & 3 & 2 & 1 \end{pmatrix}$$

2.

Scrieți o funcție care primește un număr variabil de liste și o valoare x și furnizează toate listele care conțin valoarea x .

3.

a) Pentru un student se cunosc următoarele informații: numele, grupa și o listă cu creditele obținute la toate examenele din anul respectiv. Considerând o listă de studenți, scrieți o funcție care să adauge la fiecare student situația sa școlară: promovat (True) sau nepromovat (False). Pentru a fi considerat promovat, un student trebuie să nu aibă nici un examen nepromovat (adică o valoare egală cu 0 în lista creditelor), iar suma creditelor obținute să fie mai mare sau egală decât un număr minim de credite dat.

b)

4.

Scrieți câte o funcție care să furnizeze cheia necesară sortării listei cu informații despre studenți în raport de fiecare dintre următoarele criterii:

- crescător după grupă și în fiecare grupă în ordine alfabetică;
- întâi studenții promovați, apoi cei nepromovați și în fiecare categorie în ordine alfabetică;
- descrescător după suma creditelor, iar în cazul unor sume egale în ordinea crescătoare a grupei și în ordine alfabetică în cadrul grupei;
- în ordinea crescătoare a grupelor, în cadrul fiecărei grupe mai întâi studenții promovați, iar apoi cei nepromovați, în fiecare categorie (promovat/nepromovat) în ordinea descrescătoare a sumei creditelor și, în cazul unor sume egale, în ordine alfabetică.

Seminar 4

① a)

$$M = \begin{pmatrix} 1 \\ 2 & 15 \\ 3 & 10 & 15 \\ 4 & 3 & 2 & 1 \end{pmatrix}$$

def matrice(m):

$$M = \llbracket [k+1] [0] * k \text{ for } k \text{ in range}(0, m-1) \rrbracket + \llbracket [k] \text{ for } k \text{ in range}(m, 0, -1) \rrbracket$$

for i in range(m-2, -1, -1):

for j in range(1, i+1):

$$M[i][j] = M[i+1][j-1] + M[i+1][j] + M[i][j-1]$$

return M

b) def afisare(M):

maxim = max([max(linie) for linie in M])

cifre = len(str(maxim))

for i in range(0, len(M)):

for j in range(0, i+1):

print(str(M[i][j]).rjust(cifre+1), end=" ")

print()

main:

afisare(matrice(int(input("m="))))

②

def cautare(x, *liste):

return [lista for lista in liste if x in lista]

③

a) t = ("Popa Iam", 132, [5, 4, 9, 6, 4]) → False

CSV = sistem text în care info. sunt memorate pe linii, iar pe o

linie sunt despărțite între ele prin "s"

→ ~~COMMA~~ SEPARATED VALUES

studenti.csv → Popa Iam, 132, 5, 4, 9, 6, 4

Ion Amea, 131, 10, 4, 6, 5

+ O funcție care ~~citeste~~ primește ca parametru numele unui fișier csv având structură / formă indicată și furnizează o listă de tupleuri în care fiecare tuplu conține info despre un student.

```
def citire_fisier (nume_fisier):
```

```
    f = open (nume_fisier)
```

```
    lista = []
```

```
    for line in f:
```

```
        aux = line.split(",")
```

```
        t = (aux[0], int(aux[1]), [int(x) for x in aux[2:]])
```

```
        lista.append(t)
```

```
    f.close()
```

```
    return lista
```

```
b) def promovare (lista, x):
```

```
    for t in lista range(len(lista)):
```

```
        if 0 not in lista[k][2] and sum(lista[k][2]) > x:
```

```
            lista[k] = lista[k] + (True,)
```

```
        else:
```

```
            lista[k] = lista[k] + (False,)
```