

# Problema #1

```
n = int(input("n: "))
numere = []
for _ in range(n):          # _: nu avem niciun motiv sa retinem "pasul" intr-
    o variabila (nu-l folosim la nimic) asa ca folosim "_"
    numar_nou = int(input("introdu nr: "))
    numere.append(numar_nou) # adaugam numarul nou la finalul listei

numere = sorted(list(set(numere))) # set() transforma lista numere intr-
    o multime, eliminand repetitiile
print("a)", end=" ")
for n in numere:
    print(n, end=" ")      # argumentul optional end specifica ce string sa fie la fin
alul a ceea ce printam.   # el e egal cu "\n" (new line) by default, dar cum vrem sa
                           # printam pe aceeasi linie, punem un spatiu
print()                   # printam o linie noua

L = [nr for nr in numere if nr < 0]
print("b)", L)

T = tuple(filter(lambda x: 100 <= x <= 999, numere)) # functia lambda de aici ia un
    singur argument (x) si returneaza True daca x este intre 100 si 999 inclusiv
                                                # folosim aceasta functie pentru
u a filtra elementele listei numere si apoi transformam in tuplu lista noua rezultata
print("c)", T)
n: 6
introdu nr: -1
introdu nr: 234
introdu nr: -1
introdu nr: 72
introdu nr: -1
introdu nr: 121
a) -1 72 121 234
b) [-1]
c) (121, 234)
```

# Problema #2

**Observatie:** Pentru a rula codul, descarcati in acelasi folder `rezolvare_ex2.py` (sau dati copy paste la codul de mai jos) si `date1.txt`.

```
import pprint          # pretty print -
    il folosim ca sa printam frumos dictionare (nu e esential)

liceu = {profil : dict() for profil in ["matematica-informatica", "stiinte-ale-
naturii", "stiinte-sociologice"]}
```

```

# Linia de mai sus ii atribuie variabilei liceu un dictionar cu cheile "matematica-
informatica", ... (cele din lista)
# si cu valorile setate drept niste dictionare goale (asta semnifica sintaxa profil : d
ict() => key : value

def main():
    with open("date1.txt", "r") as file:          # deschidem fisierul date1.txt cu permis
iunea "r" -> read si retinem in variabila file
        for line in file:                        # iteram prin fiecare linie din fisier
            line = line.strip().split()          # .strip() cheleste string-
ul linie de spatii " ", new line "\n" etc ... adica de white space
            cnp = line[0]                        # .split() creeaza o lista noua rezultat
a din impartirea string-ului linie dupa fiecare spatiu " "
            nume = line[1]                      # retinem cnp, nume, ...
            prenume = line[2]
            clasa = line[3]
            media = float(line[4])
            profilul = line[5]
            tuplu_elev = (cnp, nume, prenume, clasa, media, profilul)      # punem
informatiile intr-un tuplu pt. ca e convenabil
            if clasa not in liceu[profilul]:      # daca clasa nu e o cheie in su
bdictionarul asociat profilului
                liceu[profilul]
[clasa] = [tuplu_elev] # adaugam o pereche noua key:value => clasa:
[tuplu_elev] (cheia e clasa, valoarea e lista formata din informatiile acelui elev citi
t din fisier)
            else:                                # daca cheia clasa exista deja in subdictionarul asociat
profilului
                liceu[profilul]
[clasa].append(tuplu_elev) # adaugam un nou elev (acel tuplu) la lista cu elevi
pprint.pprint(liceu)      # printam cu pretty print intregul dictionar

# b)
for profil in liceu:
    print(f"Clasele profilului {profil}:")
    for clasa in liceu[profil]:
        print(f"{clasa}: media {media_clasei(liceu[profil][clasa])}")
    print()

# c)
create_fisier(liceu, "11C")

# b)
def media_clasei(clasa: list):
    media = 0
    for elev in clasa:
        media += elev[4] # elev[4] pt. la indexul 4 se va gasi media elevului
    return media / len(clasa) if len(clasa) else 0 # puteam scrie doar "return
media / len(clasa)" dar daca clasa nu
                                                    # are niciun elev, len(clasa
) == 0 => eroare de impartire cu 0
                                                    # si mai Pythonic, mergea sc
ris "... if clasa else 0"

```

```

# c)
def creare_fisier(scoala: dict, clasa: str):
    for profil, clase in scoala.items():          # iteram prin dictionar cu key:
        value fiind profil:clase
            elevi = clase.get(clasa)              # luam valoarea din dictionar a
        sociata cheii clasa (sau None daca nu exista aceasta cheie)
            if elevi:                             # daca exista acea cheie, elevi nu va fi None deci vom intra in acest if
                with open(f"{clasa}.txt", "w") as file: # de data asta deschidem un fisier cu permisiunea "w" de la write pt. ca vrem sa scriem in el
                    for elev in elevi:              # iteram prin lista de elevi
                        cnp = elev[0]
                        nume = elev[1]
                        prenume = elev[2]
                        media = str(elev[4])
                        file.write(f"{cnp} {nume} {prenume} {media}\n") # scriem in fisier ce ni s-a cerut

# d) vom rezolva acest subpunct impreuna cand veti ajunge la capitolul de expresii regulate

main()

```