

Recomandari:

- incercati sa realizati programele din cat mai puțini pași. Dacă vi se pare ca sunt prea multe for-uri, poate chiar asa e.. :-) Dar nu cadem nici in extreme cealalta și inghesuim codul doar ca sa il facem mai mic.
- puneti-va în locul cuiva (un eventual colaborator) care va trebui sa lucreze pe codul scris de voi. Dați nume cât mai sugestive variabilelor, functiilor, etc.. ca sa fie cat mai clar ce face codul.
- toate mesajele pe care le tipariti cu print trebuie sa fie cat mai clare și sugestive. Ganditi-va ca programele voastre vor fi utilizate de oameni care nu au habar de programare și trebuie sa fie cat mai clar pentru ei ceea ce afiseaza programele.
- dacă aveți nevoie de structuri de date ajutătoare (ca sa reorganizati în ele datele) ca sa rezolvati o problema, creati-va cate dictionare, liste, etc.. aveti nevoie. Nu exista nicio restrictie în acest sens.
- în codul unei funcții putem fără probleme sa apelam o alta functie definita în programul nostru
- refolositi cod cât de mult se poate: funcții definite la o problema se pot refolosi și la altele și chiar funcții folosite la temele trecute.

Avem următoarea structură de date sub forma de catalog de elevi. Cheia este numele elevului iar valoarea este lista notelor sale.

```
catalog = {  
    'Popescu Ion': [2, 5, 7],  
    'Ionescu Geta': [10, 7, 9, 7],  
    'Georgescu Gelu': [4, 2],  
    'Radulescu Ioana': [5, 9, 6, 4, 10]  
}
```

1. Scrieți o funcție care adauga o nota la un elev. Functia are doi parametri: **elev** și **nota**. Functia ii adauga la lista de note a elevului nota data ca parametru. Functia trebuie sa verifice urmatoarele:

- nota data ca parametru sa fie între 0 și 10, altfel sa dea mesaj de eroare și sa nu adauge nota
- elevul dat ca parametru sa existe în catalog, altfel sa dea mesaj de eroare

2. Mai adaugati un parametru funcției de la punctul anterior numit **overwrite**, cu o valoare default False
 - daca overwrite este False functia va functiona exact cum e descris la punctul anterior
 - daca overwrite este True atunci nota data ca parametru va inlocui toate celelalte note existente. Practic elevul va avea în acest caz o singura nota în lista și anume nota data ca parametru
3. Scrieti o functie care sterge note, cu doi parametri **elev** și **nota** iar nota sa aiba o valoare default None.
 - daca nota are o valoare diferita de None se va cauta nota în lista de note a elevului și daca exista se va sterge. Daca are mai multe note de 7 de exemplu, se va sterge oricare, nu conteaza. Daca nu exista nota, se va afisa un mesaj de eroare sugestiv.
 - daca nota este None, vor disparea toate notele elevului și va avea o lista de note goala.
4. Scrieti o functie care calculeaza media generala a clasei. Practic aceasta este media tuturor mediilor elevilor. Încercați sa refolositii funcția de calcul a mediei de data trecută.

Structura catalogului a devenit mai complexă. Acum fiecare elev are notele structurate pe materii (intr-un dictionar) cum se vede în exemplul de mai jos. **m** - matematica, **f** - fizica, **r** - limba romana

```
catalog = {  
    'Popescu Ion': {  
        'm': [2, 5, 7],  
        'f': [],  
        'r': [6, 9, 8],  
    },  
}
```

```

'Ionescu Geta': {
    'r': [6, 3, 8],
    'm': [4, 5],
    'f': [7, 9, 10]
},
'Georgescu Gelu': {
    'm': [2, 5, 7, 9],
    'r': [9, 8],
    'f': [6, 9]
},
'Radulescu Ioana': {
    'm': [7],
    'f': [],
    'r': [6, 9, 8],
},
}

```

5. Scrieti o functie de calcul a mediei, care are doi parametri **elev** și **materia**. Materia are ca valoare default None.
- la apelul functiei, daca **materia** are vreuna dintre valorile **m**, **r**, sau **f** se va calcula media elevului la acea materie și se va afisa un mesaj sugestiv de genul: **Elevul X are media la matematica 7.55**
 - daca valoarea parametrului materia este None atunci se vor calcula mediile pentru fiecare materie și se va afisa mesajul:
Elevul X are urmatoarele medii:
matematica: 7.55
limba romana: 6.78
fizica: 5.60