

Examenul scris FP:

1. Choose the correct answer:

```
class A():
    def f(self, a):
        a=a+1
    def h(self, l1,l2):
        a=l1[0]
        f(a)
        l1[0]=a
        l2=l2+l1

a=A()
l1=[1]
l2=[]
a.h(l1,l2)
print l1,l2
```

Trebuie sa scri ce se afiseaza. Nu imi amintesc sigur daca asa era codul dar ce trebuie sa ti minte: poti defini o clasa far def `__init__(self)` dar trebuie sa pui self la functii, variabilele dintr-o clasa sunt locale fiecarei functii si nu pot fi luate decat cu get-ere.

Raspunsul este: print [1][]

2. Sa scri specifications si test function la un functie. Nu mi-o amintesc dar era foarte simpla.
3. Sa calculezi time and space complexity pentru o functie.

```
def f(n)
    s=0
    for i in range(0,n*n+1):
        while i>0:
            s=s+i
            i=i-1
    return s
```

Time complexity: $T(n)=n^4$

Space complexity: Nu am stiut ce este si inca nu stiu, look up :p.

4. Ai o lista de numere. Foloseste Divide et Impera pentru a calcula numarul de numere prime din acea lista. Sa se scrie doar functia cu test si specificatii.
5. Alege una dintre metodele urmatoare care este cea mai potrivita pentru a rezolva problema de mai jos. Metode: Backtracking, Divide et Impera, Greedy, Dynamic Programming. Trebuie specificat de ce metoda aleasa este cea mai buna pentru problema data si sa se explice modul de rezolvare. (fara cod)

Pb: Ai o lista de numere. Sa se afle cea mai lunga secventa de numere pare care sunt in ordine descrescatoare.

Ex: 52,13,16,50,15,18,21 → 52,50,18