

## Exemplu rezolvare program v5 de data trecuta:

```
import random
import json

class Student:
    nrDiscipline = 3 # variabila de clasa

    def __init__(self, nume, sp, an, note):
        self.nume = nume
        self.specializare = sp
        self.anStudiu = an
        self.note = note
        self.medie = self.getMedie()

    def afisare(self):
        print("Nume: ", self.nume, "| Specializare: ",
self.specializare, "| An de studiu: ", self.anStudiu, "| Note: ",
self.note, "| Media: ", self.medie)

    def getStudent(self):
        student = {
            "nume": self.nume,
            "specializare": self.specializare,
            "an": self.anStudiu,
            "note": self.note,
            "medie": self.medie
        }
        return json.dumps(student)

    def setNume(self, n):
        self.nume = n
```

```

def setSpecializare(self, sp):
    self.specializare = sp

def setAn(self, a):
    self.anStudiu = a

def setNote(self, note):
    self.note = note

def adaugaNote(self, nota):
    self.note.append(nota)
    self.medie = self.getMedie()

def getMedie(self):
    suma = 0
    for i in self.note:
        suma += i
    return round(suma / len(self.note), 2)

def verificareRestantier(self):
    if len(self.note) < self.nrDiscipline:
        return self.numere # daca studentul nu are suficiente note
        va fi considerat restantier
    for i in self.note:
        if i < 5:
            return self.numere # daca studentul are o nota sub 5 va
            fi considerat restantier

def afisare(student):
    return "Nume: " + student['nume'] + " | Specializare: " +
    student['specializare'] + " | An de studiu: " + str(student['an']) + "
    | Note: " + str(student['note']) + " | Media: " + str(student['medie'])

def editareStudent(student):

```

```
print("Studentul ales este: ")
student.afisare()
print("1. Adauga o nota")
print("2. Modifica nume")
print("3. Modifica an")
print("4. Modifica specializarea ")
print("5. Inapoi")

op3 = int(input("Alegeti o optiune: "))

if op3 == 1:
    nota = int(input("Nota adaugata: "))
    student.adaugaNote(nota)

elif op3 == 2:
    nume = input("Noul nume al studentului: ")
    student.setNume(nume)

elif op3 == 3:
    an = input("Noul an de studiu al studentului: ")
    student.setAn(an)

elif op3 == 4:
    specializare = input("Noua specializare a studentului: ")
    student.setSpecializare(specializare)

elif op3 == 5:
    meniuCautare()

else:
    print("Optiune incorecta")
    editareStudent(student)

editareStudent(student)
```

```

def meniuEditare():
    student = meniuCautare()

    if student is None:
        print("Incercati din nou!")
        meniuEditare()

    else:
        editareStudent(student[0])

def meniuCautare():
    print("1. Cauta student")
    print("2. Selectare din lista")
    print("3. Iesire")

    op2 = int(input("Alegeti o optiune: "))

    if op2 == 1:
        s = input("Introduceti numele studentului: ")
        studenti_json = []
        for i in studenti:
            studenti_json.append(json.loads(i.getStudent()))

        for i in range(len(studenti_json)):
            if studenti_json[i]['nume'].lower() == s.lower():
                return studenti[i], i

        print("Studentul cautat nu a fost gasit!")
        return None

    elif op2 == 2:
        for i in range(len(studenti)):

```

```

        print(str(i) + ". " +
afisare(json.loads(studenti[i].getStudent()))

    s = int(input("Alegeti un student: "))
    if len(studenti) < s < 0:
        print("Student incorect")
        return None
    else:
        return studenti[s], s

elif op2 == 3:
    meniuPrincipal()

else:
    print("Optiune incorecta")
    meniuCautare()

def meniuPrincipal():
    print("1. Schimbare nr. discipline")
    print("2. Adaugare student")
    print("3. Afisare lista studenti ")
    print("4. Afisare lista restantieri")
    print("5. Modificare student")
    print("6. Stergere student")
    print("7. Afisare clasament studenti dupa media notelor, ordonati
descrescator")
    print("8. Afisare lista restantieri ordonati alfabetic dupa nume")
    print("9. Lista studentilor sa apara intr-un format tabelar la
alegere")
    print("10. Iesire")

    op = int(input("Alegeti o optiune: "))

    if op == 1:
        nr_discipline = int(input("Numar discipline: "))

```

```

Student.nrDiscipline = nr_discipline

elif op == 2:
    nume = input("Numele studentului: ")
    specializare = input("Specializarea studentului: ")
    an = int(input("Anul de studiu: "))
    nr_note = int(input("Numar note: "))
    note = []
    for i in range(nr_note):
        nota = int(input("Nota {}: ".format(i)))
        if 0 < nota <= 10:
            note.append(nota)
        else:
            print("Nota invalida")
    studenti.append(Student(nume, specializare, an, note))

elif op == 3:
    studenti_json = []
    if len(studenti) > 0:
        for i in studenti:
            studenti_json.append(json.loads(i.getStudent()))

        studenti_json.sort(key=lambda l: (l['specializare'],
l['nume']))
        for i in range(len(studenti_json)):
            print(str(i) + ". " + afisare(studenti_json[i]))
    else:
        print("Nu exista studenti inregistrati")

elif op == 4:
    if len(studenti) > 0:
        restantieri = []
        for i in studenti:
            r = i.verificareRestantier()
            if r:

```

```

        restantieri.append(json.loads(i.getStudent()))

    if len(restantieri) > 0:
        restantieri.sort(key=lambda l: (l['specializare'],
l['nume']))
        print("Restantieri: ")
        for i in range(len(restantieri)):
            print(str(i) + ". " + afisare(restantieri[i]))
    else:
        print("Nu exista studenti restantieri")
else:
    print("Nu exista studenti inregistrati")

elif op == 5:
    if len(studenti) > 0:
        meniuEditare()
    else:
        print("Nu exista studenti inregistrati")

elif op == 6:
    if len(studenti) > 0:
        student = meniuCautare()

        if student is None:
            print("Incercati din nou!")
            meniuPrincipal()

        else:
            del studenti[student[1]]
            print("Studentul a fost sters cu succes!")

    else:
        print("Nu exista studenti inregistrati")

elif op == 7:

```

```

if len(studenti) > 0:
    studenti_json = []
    for i in studenti:
        studenti_json.append(json.loads(i.getStudent()))

    studenti_json.sort(key=lambda l: (l['medie']),
reverse=True)
    for i in range(len(studenti_json)):
        print(str(i) + ". " + afisare(studenti_json[i]))
else:
    print("Nu exista studenti inregistrati")

elif op == 8:
    if len(studenti) > 0:
        restantieri = []
        for i in studenti:
            r = i.verificareRestantier()
            if r:
                restantieri.append(json.loads(i.getStudent()))

        if len(restantieri) > 0:
            restantieri.sort(key=lambda l: (l['nume']))
            print("Restantieri: ")
            for i in range(len(restantieri)):
                print(str(i) + ". " + afisare(restantieri[i]))
        else:
            print("Nu exista studenti restantieri")
    else:
        print("Nu exista studenti inregistrati")

elif op == 9:
    if len(studenti) > 0:
        studenti_json = []
        for i in studenti:
            studenti_json.append(json.loads(i.getStudent()))

```



```

de studiu      print("      Nume      |      Specializare      |      An
               |      Note      |      Media      |")
-----      print("      -----      |      -----      |      ----
               |      ----      |      ----      |")

    for i in studenti_json:
        note_joined = ",".join(str(x) for x in i['note'])
        print(" " + i['nume'] + (" " * (18 - len(i['nume']))) +
"| " +
                " " + i['specializare'] + (" " * (21 -
len(i['specializare']))) + "| " +
                " " + str(i['an']) + (" " * (21 -
len(str(i['an'])))) + "| " +
                " " + note_joined + (" " * (13 - len(note_joined)))
+ "| " +
                " " + str(i['medie']) + (" " * (14 -
len(str(i['medie'])))) + "|")
    else:
        print("Nu exista studenti inregistrati")

elif op == 10:
    return

else:
    print("Optiune incorecta")

menuPrincipal()

menuPrincipal()

```