Lab 6.

Topic: Object Oriented Programming

Task: Implement a class with the name Course_grade. The class should handle all OsloMet course grading with the following characteristics:

- A course consist of normal assignments (any number, result is accepted or rejected), graded assignments (any number, the result is a percentage) and a final exam (the result is a percentage)
- For the final exam and the graded assignments the class should be able to store weights, for the final exam there is one weight, for the graded assignments there is a list that contains the weights

The class has to provide public methods to

- Set the result of one assignment
- Set the result of all assignments at once
- Set the result of one graded assignment
- Set the results of all graded assignments at once
- Set the result of the final exam

There is no need to check invalid inputs for the set methods.

The class has to contain a get_grade public method that return the evaluated final grade considering the following logic:

- If an assignment was not submitted then the final grade is F
- If the sum percentage of the graded assignments is less than 40 percent then the final grade is F
- If the final exam percentage is less then 40 percent then the final grade is F
- In all other cases the final exam and the graded assignments are added and the final grade is evaluated using the normal grading scale (<40% F, >=40 and <50 E, >=50 and <60 D, >=60 and <80 C, >=80 and <90 B, >=90 and <=100 A)

Implement two Child classes using Course_grade as a parent with the name ACIT4420_2020 and ACIT4420_2019. The two classes should only override the constructor of the parent class with the appropriate weights. ACIT4420_2020: no assignments, 7 graded assignments (weights: 5%, 7%, 10%, 6%, 7%, 8%, 7%), final exam (50%). ACIT_4420_2019: 7 assignments, 0 graded assignments, final exam (100%).

The code and the result using the defined classes should look like the following:

```
a = ACIT4420_2020()
a.set_graded_assignments([70,70,70,70,70,70,70])
a.set_graded_assignment(2,100)
a.set_exam(76)
print(a.get_grade())

a2 = ACIT4420_2019()
a2.set_assignments([True,True,True,True,True,True,True])
a2.set_exam(76)
```

```
print(a2.get_grade())

b = ACIT4420_2020()
b.set graded assignments([20,20,20,20,20,20,20])
b.set_graded_assignment(2,100)
b.set_exam(84)
print(b.get_grade())

b2 = ACIT4420_2019()
a2.set_assignments([True,True,True,True,True,True,True])
a2.set_exam(84)
print(a2.get_grade())
```

C

C

F

B