```
# Question 1
# Given the names and grades for each student in a class of students,
store them in a
# nested list and print the name(s) of any student(s) having the
second lowest grade.
# Note: If there are multiple students with the second lowest grade,
order their names
# # alphabetically and print each name on a new line.

students = ["Alpha", "Bravo", "Charlie", "Delta", "Gamma", "Hamilton"]
grades = ["C", "B", "D", "A", "D", "E"]
nlst = []
for a in range(len(grades)):
    nlst.append([students[a], grades[a]])


def second_lowest(lst):
    ls = []
    g = []
    for a in lst:
        g.append(a[1])

    g.sort(reverse=True)
    print("Students with second lowest score :")
    for b in lst:

        if b[1] == g[1]:
            print(b)


second_lowest(nlst)
```

```
Students with second lowest score :
['Charlie', 'D']
['Gamma', 'D']
```

```
# Question 2

# You are given a string S. Suppose a character 'c' occurs
consecutively X times in the
# string. Replace these consecutive occurrences of the character 'c'
with (X, c) in the string.


string = "aaappple pie mangoo pie cocco powder cooool"

# I will ignore just print 'c' in case X==1
def charXc(string):
    string1 = ""+string[0]
```

```
    curr = 0
    while curr < (len(string) - 1):
        X = 1
        while (string[curr] == string[curr + 1]):
            X += 1
            if (curr + 1) == len(string):
                break
            curr += 1
        if X>1:
            string1 += "("+ str(X) + ")"
            continue
        if (X == 1) :

            string1 += string[curr+1]
            curr += 1
    print(string1)


charXc(string)

a(3)p(3)le pie mango(2) pie coc(2)o powder co(4)l

# Question 3

# Write a program that takes two file names, file1 and file2 as input.
The program should
# read the contents of file1 line by line and should write them to
file2, adding a newline at
# the end of each line. Note: You are required to handle all the
possible exceptions. (7
# marks)


try:

    file1=open("a2_1.txt","w")
    file1.write("This is random text I am typing in midnight\nLine2\
nLin3\nLine4\nLine5")
    file1.flush()
    file1.close()
    file1=open("a2_1.txt","r")
    f=file1.readlines()
    file2=open("a2_2.txt","w")
    data=[f[x]+"\n" for x in range(0,len(f)) ]
    file2.writelines(data)

    file2.flush()
    file2.close()
```

```
    print("Alternative Lines copied Successfully")
except:
    print("Error Occured")
```

```
Alternative Lines copied Successfully
```

```
# QUESTION 4
# . Write statement(s) to:
# a. create a new list lst2 from an existing list lst1 copying only
the odd indexed
# elements.
# b. determine the second largest element from a tuple myTup.
# c. display names of only 'red' coloured fruits, given the
dictionary:
# fruits = {'apple': 'red', 'mango': 'yellow', 'orange': 'orange',
'cherry': 'red'}.
# d. display only those words of a sentence that have five or more
characters.

#a.
lst1=[1,2,3,4,5,6,7,8,9]
lst2=[y:=lst1[x] for x in range(0,len(lst1),2) ]
print("lst2 : ",lst2)
print("----------------------------------------")
#b.
myTup=(13,10,40,2,87,9)
myTup=list(myTup)
myTup.sort()
myTup=tuple(myTup)
print("Second Largest elements from a tuple myTup : ",myTup[-2])
print("----------------------------------------")

#c.
fruits = {'apple': 'red', 'mango': 'yellow', 'orange': 'orange',
'cherry': 'red'}

for a in fruits:
    if "red"==fruits[a]:
        print(a)
print("----------------------------------------")


#d.
sentence="This is a demo sentence and holds no significance as apples
aren't spacecraft's"
sentence=sentence.split(" ")
for b in sentence:
```

```
    if len(b)>4:
        print(b)
```

```
lst2 :  [1, 3, 5, 7, 9]
----------------------------------------
Second Largest elements from a tuple myTup :  40
----------------------------------------
apple
cherry
----------------------------------------
sentence
holds
significance
apples
aren't
spacecraft's
```

```
# Question 5
#  Define a class Student that keeps a record of students. The class
should contain the
# following:
# Note: marks should lie between 0 and 100.
# data members for every student: Name, Class, rollNo, marks1, marks2,
and marks3.
# function members:
# a. __init__ method to initialize the members
# b. function grade, that returns the grade of the student according
to the following criteria:

# A : if percentage >= 90
# B : if percentage >= 70 and < 90
# C : if percentage < 70

# 6. Create an object S1 of this class with name Ashish of B.A.
(Prog.) Sem I having roll no as
# 123 and marks in three subjects as 93, 67, 70.
# 7. Display the grade of the student created in part (i).
```

```
class Student():
    def __init__(self):
        self.Name="NoName"
        self.Class="NoClass"
        self.rollNo="0"
        self.marks1=0
        self.marks2=0
        self.marks3=0
```

```
    def __init__(self,Name,Class,rollNo,marks1,marks2,marks3):
        for b in [marks1,marks2,marks3]:
            if not (0<=b<=100) :
                raise Exception("marks must be in range 0 to 100")
        self.Name=Name
        self.Class=Class
        self.rollNo=rollNo
        self.marks1=marks1
        self.marks2=marks2
        self.marks3=marks3
    def grade(self):
        percentage=(self.marks1 + self.marks2 + self.marks3)/3
        if percentage>=90:
            return "A"
        elif 90>percentage>=70:
            return "B"
        else :
            return "C"

stu1=Student("Ashish","B.A.(prog)","123",93,67,70)
print(stu1.grade())
```

```
B
```