Lab 4 - Tuple

P1. Which among the following statements may result in an error? Assume that the statements are executed in the order in which it is written.

a. tup1=(5,10,15,20,25)
b. print(len(tup1))
c. print(tup1[4])
d. print(tup1[5])
e. print(tup1[4:5])
f. tup1[2]=12
g. print(tup1)
h. tup1=tup1+(8,9)

In the statements above, statement (d) and statement (f) will give the error.

P2. Pure Gems Store sells different varieties of gems to its customers.
Emerald, Ivory, Jasper, Ruby, Garnet and their prices are 1760, 2119, 1599, 3920, 3999 Respectively. Write a Python program to calculate the bill amount to be paid by a customer based on the list of gems and quantity purchased. Any purchase with a total bill amount above Rs.30000 is entitled for 5% discount. If any gem required by the customer is not available in the store, then consider total bill Amount To Be -1. Assume that quantity required by the customer for any gem will always be greater than 0. Perform case-sensitive comparison wherever applicable.

```
gems = ("Emerald","Ivory", "Jasper", "Ruby", "Garnet")
prices = (1760,2119,1599,3929,3999)
print("The gems avalable are: ")
a = 1
for i in gems:
        print(a,i)
        a += 1;
inp = ()
amt = ()
inp = input("Enter the gems wanted: " ).split()
amt = input("Enter their amount: ").split()
bill = 0
for i in inp:
        if i not in gems:
                bill = -1
```

Aman Singh Rawat(21B056)

```
            break
        else:
        bill += (int(amt[inp.index(i)])*prices[gems.index(i)])

if bill >30000:
        print("Your puchase is applicable for 5% discoutnt: ")
        print("Your total is: ",(bill - bill*(5/100)))
else:
        print("Your bill is: ", bill)
```

Output:

```
The gems avalable are:
1 Emerald
2 Ivory
3 Jasper
4 Ruby
5 Garnet
Enter the gems wanted: Emerald Ivory Jasper
Enter their amount: 12 23 48
Your puchase is applicable for 5% discoutnt:
Your total is:  139278.55
```

P3. Write a python function to check whether three given numbers can form the sides of a triangle.

```
def check_triangle(a,b,c):
        if (a <= b+c) and (b <= c+a) and (c <= a+b):
                print("Triagle is possible.")
        else:
                print("Triangle is not possible.")

a = ()
a = input("Enter the sides: ").split()
check_triangle(int(a[0]), int(a[1]), int(a[2]))
```

Output:

```
Enter the sides: 3 2 2
Triagle is possible.
```

P4. Execute the program triangle.py available in the folder. The program triangle.py is

Aman Singh Rawat(221B056)

written to display "*" as per the expected output given below. But the code is having logical errors, debug the program and correct it.

```
counter1 = 0
while counter1 < 5:
        star = ""
        counter2 = 5
        while counter2 > counter1:
                star = star+"*"
                counter2 -= 1
        print(star)
        counter1 += 1
```

Output:
*****
****
***
**
*

P5. Write a python program to solve a classic ancient Chinese puzzle.
We count 35 heads and 94 legs among the chickens and rabbits in a farm. How many rabbits and how many chickens do we have?

```
def solve(heads,legs):
        error_msg="No solution"
        chicken_count=0
        rabbit_count=0
        if(heads>=legs):
                print(error_msg)
        elif(legs%2!=0):
                print(error_msg)
        else:
                rabbit_count=(legs-2*heads)/2
                chicken_count=heads-rabbit_count
                print(int(chicken_count),int(rabbit_count))
solve(38,131)
```

Output:
No solution

Amar Singh Rawat(221B056)