Quiz-1: CSE101 Introduction to Programming (Set-A)

Name: **Roll No:** **Section:** **Group:**

Note:

- *Time: 30 mts. Each question carries 1 mark. Best 10 answers will be counted.*
- *This is an exam - use of unfair means will be handled as per institute policy.*
- ***Give answers in the space provided. Return the sheet at the end of the quiz.***
- ***Don't forget to write your name and roll number.***

Q1. What will be the output of the code given below?

```
def my_test(a, b):
    x = a + b
    y = x * b
    z = y + b
    print("Inside Function:", x, y, z)
    return x, y

a = 2
b = 3
z = 1
y, x = my_test(b, a)

print("Outside Function:", x, y, z)
```

Output:

Q2. What will this function return:

```
def f():
    num_list = [1,2,3,4,5,6,7,8,9,10]
    p = -1
    result = 0
    for num in num_list:
        result = result + num * p
        p = p * -1
    return result
```

Output:

Q3. Evaluate the following expression:
(x >= 5 or x>=y) and (y > 10 or y>z)
for x, y, z = 5, 10, 10

Output:

Q4. The fizz_buzz function is given a natural number n as a parameter. The function should loop through all the numbers from 1 to n in turn. When the number is divisible by 3, the function prints “fizz”; when it is divisible by 5, it should print “buzz”. If the number is divisible by both 3 and 5, it should print “fizz_buzz”. Else, the number is returned itself.

Fill in the blanks below so that the function behaves as mentioned in the above problem statement (Give your answer next to the blank:

```
def fizz_buzz(n):
      for i in range(n):
        (b) ____  i%15==0:
                  print("fizzbuzz")

        (c) ____ i%3==0:
                  print("fizz")

        (d) ____ i%5==0:
                  print("buzz")

        (e) ____:
                  print(i)
```

Q5. if we call the function as: nested(10) - how many times will "outer"be printed and how many times will "inner" be printed.

```
def nested(n):
    for i in range(n):
          print("Outer")
          for j in range(i):
               print("Inner")
```

Ans:
# of times "Outer" is printed:
# of times "Inner" is printed:

Q6. Given an AP 121, 117, 113, ……. We have to write a program to determine the first negative term of the AP and its place in the series i.e. say if a_n is the first negative term, find the value of a_n and n. The code for this is given, with some missing parts (marked as ?). Fill the missing parts (write your answer next to ? itself)

```
def find_number(first_term, d):
    n = 1
    required_term = ?
    while(required_term ?):
        required_term = ?
        n += 1
    return required_term, n

first_term = 121
second_term = 117
d = second_term - first_term
a_n, n = find_number(first_term, d)
```

Q7. Consider the following code.

```
def f(x):
    return x*2

def compose(f, g, x):
    return f(g(x))

def h(f, x, n):
    result = x
    for i in range(n):
        result = compose(f, f, result)
    return result
```

Given x = 2 and n = 3, **what will h(f, x, n) return:**

Q8. Briefly explain what the function 'h' (given in Q7) is effectively computing.

Ans:

Q9. In a short sentence describe what this function is doing.

```
def fn(N, a, b):
   count = 0
   for i in range (N):
      if i%a == 0:
         if i%b == 0:
            count += 1
   return count
```

Ans:

Q10. Given two function definitions fn1 and fn2 which return a float value, as:

```
def fn1 (x):
      …
def fn2(x):
      …
```

In the main program, to compute the product of the values of these functions for a given value of x, it calls a function. The main program is:

```
x = int(input("x: "))
z = compute(fn1, fn2, x)
print(z)
```

Write the code for the function **compute:**

Q11: How many times will the loop be executed in this program:

```
sum = 0
n = 1
while (n>=1):
        n += 1
        sum += n
print(sum)
```

Ans:

Q12: Want to write a program to repeatedly take integer input x from the user and find the sum of these inputs. Program stops taking inputs whenever a negative value is given as input (this value is discarded for sum). Write one statement (incl conditionals, loops, …) in place of ? in each of these solutions.

```
sum=0
while (True):
        ?
        ?
print(sum)
```