Q1. Solution
**Inside Function: 5 10 12**
**Outside Function: 10 5 1**

Q2. Solution: **5**

Q3 Solution: **False**

Q4 Solution: **(b) if**
        **(c) elif**
        **(d) elif**
        **(b) else**

Q5 Solution:
Ans: **(i) 10**
    **(ii) 45**
**The outer loop runs with values of i ranging from 0 to n-1, and thus, "Outer" is printed 'n' times.**
**The inner loop is nested in thus, for every 'i', it runs with the value of i ranging from 0 to (i - 1).**
**Thus effectively running 0 + 1 + 2 … + (i - 1) times.**

Q6 Solution:
Solution:
```
def find_number(first_term, d):
    n = 1
    required_term = first_term
    while(required_term > 0): //give full marks if required_term>=0 or
required_term>0 is written
        required_term = required_term + d
        n = 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 Solution:
**Solution: 128**

Q8 Solution:
**The function h(f, x, n) is effectively computing:**

        **f(f(f(.........2n times (x)))**

**which is equal to  $x * 2^{2n}$**

**Thus, for x = 2 and n = 3, value of h(f, x, n) is 128**

Q9 Solution:
**Determining the number of values in range 0..N that are divisible by two a and b**

Q10 Solution:
**def compute(f, g, x):**
      **return(f(x)\*g(x))**

Q11 Solution:
**Ans: Infinite loop (it will run forever)**

Q12 Solution:
**Ans:**
**x=int(input("x: "))**
**sum+=x**