Homework 7 Solutions

Ans.1) initial passed value of counter

Ans.2) void print(int n)

```
{
   if (n==0) return;
   print(n-1);
   cout<<n;
   return;
}
```

Ans.3) i. num-1

ii. num

Ans.4):

3 2 1

no of function call=4(for 3,2,1,0)

5.Ans)

output 6

No. of function calls=5.

Modified program(Changes are in bold)

```
#include<simplecpp>
int count=0;                //count is declared as a global variable. So all increment will update it
main_program{
        int f( int n);
        int a=f(5);
        cout<<a<<endl<<"Number of function calls="<<count;
}
```

```
int f(int n){
        int r=0;
        if (n<=0)
        {
                count++;
                return 1;
        }
        if (n>3){
                r=n;
                count++;
                return f(n-2)+r;

        }
        else
        {
                count++;
                return f(n-1)+r;
        }
}
```

This count++ is always be before the return statement, becoz if we write it after the return statement, it will not be execute.

Ans.6) Recursion will not terminate if a<0.

For a>=0, func(a, b) will return the value of a+b, and the function will be called exactly a+1 times.

7. Ans)

a) 0

b) p-1

8. Ans) a- i+1

Ans.9) i. &a

ii. &b

iii. int * x

iv. int * y

v. *x

vi. *x
vii. *y
viii. *y

10.Ans) 110

11. Ans)

a. Valid, becoz our array is declared of size 10, but only few values assigned to it, so remaining will be by default assign to zero.
b. Valid, becoz array name itself gives the base address of array (or) address of array[0].
c. Invalid, We can't change the value associated to the name of array, On declaration of array, a address values is associated to it. we do not have any control on it, becoz it is not necessary that 1000 will be free according to our required size(size of array).
d. Valid, Before declaration of array ar we know the value of x.
e. Invalid, At time of declaration of we do not know the value of n. So size of array is ambiguous.
f. Valid, Both p &q have same data type so we can assign one value to other.

12.Ans) d, func is called by reference, and then p & q both point to j. So value of j is changed by *p=2, but value of i is remain same.

13. Ans) Errors
semicolon missing in structure declaration.
seconds.t1 should be t1.seconds
output- 19845

14.Ans) No, 65

15.Ans) A = 8
B = 8
C = 27

16.Ans) 7 (The program finds the sum of the numbers less than 6)

17.Ans)
Before swap, value of a :100
Before swap, value of b :200
After swap, value of a :200
After swap, value of b :100

18.Ans) (d) None of the mentioned
We are adding all the elements in the array and printing it. Total elements in the array is 7, but our for loop will go beyond 7 and add a garbage value.

19. Ans) a,b,c
The members can be arrays, variables of other data types(including other structures).

20.Ans)
B)
Note that the type of variable a is T1.

The assignment 'p = &a' suggests that the type of p should be T1 *. The assignment 'a = *b' suggests that b is a pointer to something that has the same data-type as a. Therefore, the type of b should be T1 *. The assignment 'b = &c' suggests that the type of b should also be T3 *, since the type of c is T3.

Since the type of b should both be T1 * and T3 *, therefore, we should have T1 and T3 representing the same data-type. Finally, the assignment 'c = *d' suggests that d is a pointer to something that has the same data-type as c. Therefore the type of c should be T3 *.

Putting all the above together, we have that T1 and T3 should represent the same data-type, and T2, T4 and T5 should be a pointer to T1 (or T3).

Only the second option satisfies the above constraints. In this option, T1 and T3 are int *, while T2, T4 and T5 are int **.