Doubt: Q4

12.5/15

Quiz 2: Set B

Introduction to Computing and Programming (CSD101)

Max. Marks: 15 **Date: 24-10-2024**

Duration: 30 min.

Name: ADYA SINGHAL **Roll No.** 24101100027

Note: Provide reasoning to all the MCQ type questions

1. What is the output of the following Code snippet (1 mark)

```
#include<stdio.h>

int main()
{
int a = 10, b = 20;
int *p = &a, *q = &b;
p=q;
return 0;
}
```

Solution:

p=q assigns value of q to p, ∴ both p & q will point to b.

a. Both a and b will contain 20
b. Both p and q will point to b
c. Both p and q will point to a
d. Both a and b will contain 10
e. None of these

2. What is the output of the following program? (2 marks)

```
#include <stdio.h>
int main() {
   char c[] = "ICRBCSIT17";
   char *p = c;
   printf("%s\n", c+p[2]-p[6]-1);
}
```

Solution:

~~ICRBCSIT17~~ It will result in syntax error.

3.

A. The following statement (1 mark)

Solution:

```
if (a>b)
if (c>b)
printf("One");
```

```
else {
if (c==a) printf("Two");
else printf("Three");}
else printf("Four"); }
```

a. Results in a syntax error
b. Prints Four in c<=b
c. Prints Two if c<=b
d. Prints Four in a<=b

B. The above statement can never be printed (1 mark)

a. One
b. Two
c. Three
d. Four

Solution: "Two" will never be printed as if c = a then it will also satisfy c>b and "One" will already get printed so ~~it will NOT go to~~ the else statement will NOT be executed.

4. Choose the statements that are syntactically correct (1 mark)

a. /* Is /* this is valid */ comment*/
b. for(; ;);
c. return;
d. return(5+2);

Solution:

(b) is incorrect as the correct syntax is :-

```
for(;;) { statement;
}
```

5. What is Recursion, represent it with an example? Write two differences between Recursion and Iteration. (4 marks)

Recursion divides a problem into subproblems and uses the same function to solve it.

Example:- To compute the sum of n natural numbers.

```
# include <stdio.h>
int sum(n){
  if (n==1){
    return1; }
return (n + sum(n-1)); }
```

```
int main() {
int num;
printf("Enter number: ");
scanf("%d", &num);
int res = sum(num);
printf("%d", res);
}
```

Recursion repeatedly calls the same function whereas iteration does not call the same function.

The no. of times a recursive function ~~do~~ calls the function is not predecided and depends on the base condition. Whereas, in iteration we know the number of times iteration is gonna take place.

6. Define two functions in C (**Note**: Do not write the whole program):
   [a] A recursive function to count the number of digits for a given integer. (2.5 marks)

```
int count(num){
if (num%10==0){
   return 1;
else {
 return (1+ count(num/10));
}
}
```

Eg 123

1 + count(12)

1 + count(1)

3

2.5

[b] A function to swap two integers using 'call-by-reference'. (2.5 marks)

```
void swapnum (int *a, int *b) {
    *a = *a + *b;
    *b = *a - *b;
    *a = *a - *b;
    printf("Swapped values are %d, %d", *a, *b);
}
```

For call by reference :-

```
swapnum (&x, &y);
```

[where x & y are numbers that we want to swap]

2.5