



No. of Pages : Three
No. Of Questions : Four
Examiner : Dr. Emad Elsamahy
Dr. Ahmed Salem

CS143 Introduction to Problem Solving and Programming (Final Exam)

Question One:

Mark (✓) or (X) for each of the following sentences:

1. "Stub" is used during the system testing of your program. ()
2. You can add a multi-lines comment to your C program by using (//) at the beginning and end of your comment. ()
3. All types of loops in C can be implemented using for loop. ()
4. Binary search can be used to save searching time in case of unsorted array. ()
5. To increment a variable (i) by (1), all the following commands are valid: ()
 $i=i+1$, $i+=1$, $++i$, and $i++$
6. In C, short circuit evaluation can be used as a trap for the program logical error. ()
7. In C, variable name can be any name (satisfying the rules) or a standard identifier. ()
8. The loop in the shown program will be repeated for 5 times. ()

```
#include <stdio.h>
void main()
{
    int i=5;
    while (i>=1)
        printf("Hello\n"); --i;
}
```

9. Recursion is used as a replacement of looping in C programs. ()
10. Both (scanf) and (gets) can accept the input "Problem solving" correctly. ()

Question Two:

Choose the correct answer (a, b, c, or d):

1. The shaded range on the right can be described using: 
a. $(X \geq 0 \parallel X \leq 100)$
b. $(X \leq 0 \ \&\& \ X \geq 100)$
c. $(X \leq 0 \parallel X \geq 100)$
d. $(X \geq 0 \ \&\& \ X \leq 100)$
2. Using the method of indirect reference in C, supposing that (Y) is a pointer to (X), the following statement can be used to set the value of (X):
a. $X=5$;
b. $\&Y=5$;
c. $*Y=5$;
d. $X= \&5$;
3. After execution of $(\text{status}=\text{scanf}(\text{"\%lf", \&a});)$, and the user input (a) as 'c'. (status) should return the value of:
a. 0.
b. 1.
c. 2.
d. -1.
4. Any array element (e.g. element 4) can be incremented by (1) using the following statement:
a. $X[4]++$;
b. $X[3]+1=X[3]$;
c. $X[3]+=1$;
d. $X[4]+=1$;
5. To change the value of a variable (X) inside a function named (SetVal), the function header should be as follows:
a. $\text{void SetVal}(\text{int } X);$
b. $\text{void SetVal}(\text{int } *X);$
c. $\text{void SetVal}(\text{int } \&X);$
d. (b) and (c).
6. To prevent any change in the contents of an array variable (List) within a function, the following function header can be used:
a. $\text{get_max}(\text{const int List}[], \text{int } n);$
b. $\text{get_max}(\text{int } *List[], \text{int } n);$
c. $\text{get_max}(\text{const int List}, \text{int } n);$
d. $\text{get_max}(\text{int } *List, \text{int } n);$

7. To swap the contents of two variables, without the use of a third variable, the following C code can be used:

- a. $y=x+y;$
 $x=y-x;$
 $y=y-x;$
- b. $x=x*y;$
 $y=x/y;$
 $x=x/y;$
- c. $y=x-y;$
 $x=x-y;$
 $y=x+y;$
- d. (a), (b), and (c).

8. To left justified a string (y) when printed to the screen using (printf), the following command can be used:

- a. `printf("%10s",y);`
- b. `printf("%-10s",y);`
- c. `printf("%[10]s",y);`
- d. `printf("%-[10]s",y);`

9. For a string variable named (str[15]) which equals to "Mohamed Amr", running the following C code will change the (str) content to:

- a. "mOHAMED aMR".
- b. "MOHAMED AMR".
- c. "mohamed amr".
- d. "Mohamed Amr".

10. function can be used to append a sub-string to the end of a string.

- a. `strcat.`
- b. `strncat.`
- c. `strcpy.`
- d. (a) and (b).

Question Three:

(10 Points)

Trace the following C programs.

1 (3 Points)

```
#include<stdio.h>
#include<string.h>
void main()
{
    char s[35];
    char *a;
    int i=1;

    strcpy(s,"Introduction to problem solving");
    printf("%s\n",s);

    a= strtok(s, " ");
    printf("%s.\n",a);
    while (i<=3)
    {
        a= strtok(s, " ");
        printf("%s.\n",a);
        i++;
    }
}
```

2 (4 Points)

```
#include<stdio.h>

void main()
{
    int a[]={10,3,4,7,2,1};
    int i;

    for (i=0;i<5;i=i+2)
    {
        a[i]=a[i]+a[i+1];
        a[i+1]=a[i+1]*a[i+1];
        if (a[i]%2==0) continue;
        if (a[i+1]%7==0) break;
    }

    for (i=0;i<=5;i++)
        printf("%d\n",a[i]);
}
```

3 (3 Points)

```
#include <stdio.h> ✓
void main() ✓
{
    FILE *inp;
    int a, b, c, i=1;
    inp=fopen("Data.txt","r");
    while (i<=3)
    {
        fscanf(inp,"%d %d %d", &a, &b, &c);
        printf("%d:%dX%d=%d\n", a, b, c, b*c);
        i++;
    }
    fclose(inp);
}
```

Data.txt
1000 5 1
1001 4 2
1003 8 7

Question Four:

Write C program to solve the following problems.

(10 Points)

1. It is required to ask the user to enter the following data items for 10 students:

(4 Points)

- Student ID.
- Student GPA.

Then, it is required to print these data sorted in a descending order as follows:

Student ID	(2 tabs)	GPA
20105051		3.95 → 1
20103440		3.84 → 2

Your solution should contain the following functions:

- A function to accept the student ID and the GPA, which will be stored in *parallel arrays*. The function prototype is:

GetData(int ID[], double GPA[], int *n); ✓

- A function to sort the data using any sorting technique (e.g. bubble sort or selection sort). The function prototype is:

DataSort(int ID[], double GPA[], int n);

- A function to display the data as previously mentioned. The function prototype is:

DispData(int ID[], double GPA[], int n);

2. Using for-loop, write a program that accept a series of numbers (with unknown count) ended by zero, then compute and display the following: (3 Points)

- Sum and average of even numbers.
- Sum and average of odd numbers.

Note that: DO NOT use while or do-while loops.

3. Write a program using, a **recursive function**, to convert from decimal to binary. (3 Points)

The main program must accept the decimal number, your recursive function should convert the number into binary, and the main program should print the results.

Note that: the program must include a recursive function and using any other method rather than recursion will not be counted.

Best Wishes