Bangabandhu Sheikh Mujibur Rahman Science and Technology University
Department of Computer Science and Engineering
1st Year 1st Semester B.Sc. (Engg.) Examination-2013
Course No-CSE112 Title: Structured Programming

Full Marks:70 Time:4 hours

N.B.

i. Answer **SIX** questions, taking any **THREE** from each section.
ii. All questions are of **equal** values.
iii. Use separate answer script for each section.

Section A

1. a) What are the primitive data types used in C? State their memory consumption rate. 3.67

   b) Interpret the following program in C: 4

```
#include<stdio.h>
int main()
{
    printf("Hello, World!");
    return 0;
}
```

   c) Determine which of the following are invalid identifiers and why? 2
   (i) `record1` (ii) `$tax` (iii) `file_3` (iv) `return` (v) `student name` (vi) `1student`

   d) Differentiate logical and relational operators in C with appropriate examples. 2

2. a) Given `int n, i=10, j=20, x=3, y = 100;` 2
   What is the value of n and y at the end of the following expressions?
   `n = (i > j) && (x < ++y);`

   b) Suppose, a, b, c are integer variables with initial value 50, 10 and 20 respectively. Find the value of c after execution of the following statement: 3

```
c += ( a > 0 && a <= 10 ) ? a++ : a/b ;
```

   c) What is wrong with the following code? 3.67

```
main(){
    int *ptr;
    ptr=10;
}
```


   d) What is the difference between converting a data-type by conversion characters and by data-type casting? Illustrate with appropriate examples. 3

3. a) The length of a memory address is 32 bit. What would be the output of the following C program? 2

```
#include <stdio.h>
int main(){
char *pointer;
printf("sizeof(*pointer)=%d and sizeof(pointer)=%d",
sizeof(*pointer), sizeof(pointer) );
}
```

   b) What is the difference between call by value and call by reference? Clarify it with example of each. 3.67

   c) Describe the memory representation of a two-dimensional array. 3

   d) Illustrate the following declarations in terms of memory usage: 3
   i) `int *p=new int;` ii) `int a[3][4] ;` iii) `int *a[3];`

4. a) Write a program to test a string as a palindrome. 5

   b) Illustrate the printed result of the following program: 4

```
#include<stdio.h>
main()
{
    int u=3;
    int v;
    int *pu;
    int *pv;
    pu=&u;
    v=*pu;
    pv=&v;
    printf("%d",*pv);
}
```

   c) When is the function prototype necessary and why? 2.67

Section B

5. a) Present the difference between continue and break statement with example. 3

   b) Write the output of the following code: 5

```
#include <stdio.h>
main()
{
   int p ,q, r=0;
   for (p=0; p<5; ++p)
      for (q=0; q<p; ++q){
         r+= (p+q-1);
         printf("%d",r);
      }
}
```

   c) Convert the following switch statement into if-else equivalent statement 3.67

```
int k;
for (int i=50; i<=100; i=i+1){

   switch(i%4) {
      case 0: k=0;
              break;
      case 1: k=1;
              break;
      default:k=3;
              break;
   }
}
```

6. a) Write a simple *for* loop to print the following number sequence until crossing an integer n given by user: 4.67
   1, 1, 2, 3, 5, 8, 13, ...

   b) Discuss how union differs from structure considering suitable example. 3

   c) Differentiate local and global variable. 2

   d) How many times the program will print "Bangladesh"? 2

```
#include<stdio.h>
int main(){
   printf("Bangladesh");
   main();
   return 0;
}
```

7. a) Why is structure necessary? How can it be declared in C? 2

   b) Write a program to insert n number of students with their name, id and obtained_marks in CSE112 into a linked-list. 5

   c) What are the modes of opening a file? What is the purpose of *fclose* function? 2+1.67

   d) What is the difference between 'a' and "a"? 1

8. a) What is the base criteria of a recursive function? 1.67

   b) Illustrate the execution of a recursive function in memory to produce factorial of 5? 4

   c) Assuming, integer is of 2-byte size, what will be the output of the following program? 2

```
#include<stdio.h>
int main(){
   printf("%x\n",-1>>1);
   return 0;
}
```

   d) Write a function to reverse a string using a for loop. Now, convert the function into recursive. 3+1