Indian Institute of Technology, Kharagpur
Computer Science and Engineering Department

Class Test II, Autumn 2012-13
Programming & Data Structure (CS 11002)

Full marks: 30 **March 21, 2013** **Time: 60 mins.**

Name	Roll No.	Section	Marks obtained

1. The following sequence of six statements is taken from a C program. One among the last four statements will lead to compilation error. Write that incorrect statement in the box below. [1]

```
int *i; int j[2]={10,11}; //correct
i=j[0]+1; i=&j[0]+1; i=j; i=j+1; //one is incorrect
```

```
i=j[0]+1;
```

▶ No part marking.

2. What will be displayed when the following program executes? [2]

```
#include <stdio.h>
#include <string.h>
int main(){
  char a[10]={'p','\0','d','\0','s','\0'}, *b="p\nd\ns\n";
  printf("%d %d\n",strlen(a),strlen(b));
  return 0;}
```

```
1 6
```

▶ 1 mark each.

3. The following program has to read a string of alphanumeric characters of length at most 40, and has to reverse the string. Some parts of the program are missing (indicated by dotted line). Fill up these missing parts. [2]

```
#include <stdio.h>
#include <string.h>
#define LEN 50
int main(){
    int i, d, len; char a[LEN];
    scanf("%s", a);
```

```
    len = strlen(a)-1;

    for (i=0; ____________________________; i++){ //1 mark

        d= ______________________________; //1 mark
        a[i]=a[i]-d, a[len-i]=a[len-i]+d;}
    printf("%s\n", a);
return 0;}
```

```
i<=len/2
a[i]-a[len-i]
```

4. What will be displayed when the following program executes? [2]

```
#include <stdio.h>
static int data[10] = {2, 4, 7, 9, 12, 15, 22, 30, 42, 45};
int binarySearchComp(int data[], int lo, int hi, int key, int c) {
    while(lo != hi) {
        int mid = (lo+hi)/2;
        c++;
        if(key <= data[mid]) hi = mid;
        else lo = mid+1;
    }
    if(key == data[lo]) printf("%d, ", lo);
    else printf("-1, ");
    return c+1;
}
int main(){
    int c;
    c = binarySearchComp(data, 0, 9, 30, 0);
    printf("%d.\n", c);
    return 0;
} //binarySearchComp.C
```

```
7, 4.
```

► 1 mark each. $\frac{1}{2}$ mark should be deducted in case comma or/and dot is/are not written.

5. What will be displayed after the following code segment executes? [2]

```
double trouble = 13.0, *pt = &trouble, **ppt = &pt; //trouble.c
*pt=0;
**ppt = 13.13;
printf("trouble = %f \n", trouble);
```

```
trouble = 13.130000
```

6. What will be displayed when the following program executes on the input: `pds class test 2` [3]

```
#include <stdio.h>
#define LEN 100
int main(){
    int i=0, j=0; char a[LEN];
    scanf("%[^\n]", a); // input: pds class test 2
    do{
        if (a[i]==' '){
            i++;
            a[j++]=a[i++];
            if (a[j-1]>'Z') a[j-1]=a[j-1]-('a'-'A');}
        else a[j++]=a[i++];
    }while(a[j]!='\0');
    printf("%s\n", a);
return 0;}
```

```
pdsClassTest2
```

► Full marks if one writes `PdsClassTest2`, assuming a space before `p`. 1.5 marks if one writes `pdsclasstest2`.

7. What will be displayed after the following code is executed? [1+2]

```
#include <stdio.h> //bar.c
void bar(int M[][3]){
  printf("bar: %d\n", M[1][1]);
}
int main(){
  int M[3][2] = {{1,2}, {3,4}, {5,6}};
  printf("main: %d\n", M[1][1]);
  bar(M);
  return 0;
}
```

```
main:  4
bar:  5
```

8. Consider a computer that has 4-byte memory addresses. What will be the output of the following code segment? [3]

```
typedef struct node{ //struct-node.c
  char studName[20]; float grades[5]; char rollNo[8];
  } stud;
printf("%d %d %d\n", sizeof(char **), sizeof(stud), sizeof(float *));
```

```
4 48 4
```

► 1 mark each

9. The following program should fill up the 2D array `T` with elements of 1D array `S`, which are all distinct, such that:
– all elements of each row of `T` are distinct;
– all elements of each column of `T` are distinct.
Write the missing expression of the program (indicated by dotted line). [4]

```
#include <stdio.h>
#define MAX 5
int main(){
  int i, j, T[MAX][MAX], S[MAX]={1,2,3,4,5};
  for (i=0; i<MAX; i++)
    for (j=0; j<MAX; j++)

      T[i][j]=_____________________________________________;
  return 0;
}
```

```
S[(i+j)%MAX]
```

► No part marking.

10. What output would the following program produce? [4]

```
#include <stdio.h>
int main(){ //addnum-for.c
  int numbers[10]={1, 0, 0, 0, 0};
  int i, j;
  for (j = 0; j < 5 ; ++j)
    for (i = 0; i < j; ++i)
      numbers[j] += numbers[i];
  for (j = 0; j < 5; ++j)
    printf("%d ", numbers[j]);
  return 0;
}
```

```
1 1 2 4 8
```

► No part marking.

11. Consider a recursive function `recFunc()` defined as follows. Assume that from the `main` program, `recFunc()` is called once as `recFunc("PDS")`. What will get displayed due to the call? How many times would `recFunc()` be called? Include in your count the original call made to `recFunc("PDS")` in the `main()`. [2+2]

```
void recFunc(char str[]) { //recFunc.c
  int t;
  t = strlen(str);
  if (t == 0) return;
  if (t % 2 == 0) recFunc(&str[t/2]);
  else recFunc(&str[1]);
  printf("%s",str);
}
```

SDSPDS called 4 times

► 2 marks each. No part marking.