

INDIAN INSTITUTE OF TECHNOLOGY, KHARAGPUR  
COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

MID-SEMESTER EXAMINATION, AUTUMN 2012-13  
PROGRAMMING & DATA STRUCTURES (CS 11002)

Full marks: 100

Time: 2 hours

Name	Roll No.	Section

1. Answer all questions in the space provided in this question paper itself. No extra sheet will be provided.
2. As mentioned in class, questions have been shuffled across different question papers.
3. **No further clarifications to any of the questions will be provided.**

---

Section A: 2 Mark Questions

---

1. What will be displayed when the following program executes?

```
#include<stdio.h>
int main() {
    int i, j, k=10;
    for (i=j=k; j<=i; i=i+k)
        j=j+i;
    printf("i=%d, j=%d\n",i,j);
    return 0;}
```

i=30, j=40

2. How many times will the following `while` loop execute?

```
char a='a';
while(a > 'a' && a <= 'z') a++;
```

Ans: 0 times.

3. What is the value of `x` after the execution of the following program segment?

```
x=-5; y=10;
if(x>y) x=1;
else if(y<0) x=(x)*(-1);
else x=2*x;
```

Ans:  $x = -10$ . Both the `if` conditionals evaluate to false.

4. Convert the hexadecimal number `B5.C7` to its equivalent binary number.

Hexadecimal: `B 5 . C 7`  
Binary: `1011 0101 . 1100 0111`

▷ 50% marks, if no step is shown.

If the number is converted to decimal and then to binary, then full marks may be awarded if the result is correct.

5. What will be the value of `sum` after the following C program segment completes?

```
int sum=1, index=9;
do{
    index = index-1;
    sum = 2*sum;
}while(index > 9);
```

2

6. What will be displayed when the following code segment is executed?

```
int i=0;
for(;i<=2;)
    printf("%d \n",++i);
```

1  
2  
3

▷ No marks to be deducted in case one does not show new lines.

7. What will be the value of `x` after the following code segment executes?

```
int x = 6, y = -8;
x *= x % y > x - y ? --x : y++;
```

-48

8. What will be the value of `x` after the following code segment is executed?

```
int x = 3, counter = 0;
while ((x-1)) {
    ++counter;
    x--; }

```

1

9. What will be the value of x after the following code segment is executed?

```
int x = 3;
if(x == 2);
x = 0;
if(x == 3)
    x++;
else x += 2;

```

2

10. What will be the output produced by the following C code?

```
#include<stdio.h>
int main(){
    int i=1;
    for(i=0;i=-1;i=1){
        printf("%d",i);
        if(i!=1) break;}
    return 0;}

```

-1

11. What will be the output produced by the following C code?

```
#include <stdio.h>
int i;
void increment(int i) { i++; }

int main(){
    for(i = 0; i < 5; increment(i))
        printf("i=%d\n", i);
    return 0;}

```

0 0 0 ...  
or print 0 infinitely.

12. Consider the following code segment. In the space given below, write an **if statement** to convert the character `ch` to upper case, if the user enters a lower-case character. (Do not use any library routine)

```
char ch;  
ch=getchar();
```

```
if(ch>='a') ch -= 'a'-'A';  
or if(ch>='a') ch += 'A'-'a';  
or if(ch>='a' && ch<='z') ch -= 'a'-'A';  
or if(ch>='a' && ch<='z') ch += 'A'-'a';
```

▷ Answers based on hard-coded ASCII values of 'a' and 'A' should get 50%.

13. What will be the output produced by the following program?

```
#include<stdio.h>  
void swap(int,int);  
  
main(){  
    int i=5, j=7;  
    swap(i,j);  
    printf("%d, %d\n", i, j);}   
  
void swap(int i, int j){  
    int tmp;  
    tmp = i; j = i; i = tmp;}
```

5, 7

14. What value will the following function return when called as `recur(3)`?

```
int recur(int data){  
    int k;  
    k=(data>2)?(recur(data-1)-recur(data-2)):1;  
    return k;}
```

0

---

### Section B: 3 Mark Questions

---

15. Consider the following program segment. The function `pow(x,y)` returns  $x^y$ . What is the value of `i` after execution of the program segment?

```
#include <math.h>
int x=1, i=1;
while(x <= 1000){
    x=pow(2,x);
    i=i+1;}

```

i = 5

16. What will be printed by the following program?

```
#include <stdio.h>
void func() {
    int x = 5;
    static int y = 15;
    x++; y++;
    printf("%d\n", x*y);}

int main(){
    func();
    func();
    return 0;}

```

96  
102

17. What will be printed by the following code segment?

```
int x = 0;
for ( ; ; ) {
    if (x++ == 4)
        break;
    continue; }
printf("x=%d\n", x);

```

5

18. Convert the decimal number 101.125 to its hexadecimal representation.

Decimal to Binary: 101 → 1100101 and .125 → .001  
 Binary: 0110 0101 . 0010  
 Hexadecimal: 6 5 . 2

19. Consider the following if-else statement structure:

`if(e1) if(e2) s1; else s2; else if(e3) s3; else s4;`

where `e1`, `e2`, `e3` are logical expressions, and `s1`, `s2`, `s3`, `s4` are statement blocks. Choose the option that shows the correct matching of the else statements with the corresponding if statements.

(a) <code>if(e1) {     if(e2)         s1;} else s2; else {     if(e3)         s3;} else s4;</code>	(b) <code>if(e1) {     if(e2)         s1;     else         s2;} else if(e3)     s3; else s4;</code>	(c) <code>if(e1) {     if(e2)         s1;     else         s2;} else {     if(e3)         s3;     else s4;} }</code>	(d) The given if-else statement structure is logically impossible or incorrect.
--------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------

(b) or (c)

20. What will be printed by the following code segment?

```
for (int row = 1; row <= 3; row++){
    for (int count = 1; count <= (4 - row); count ++){
        printf("*");
        printf("\n");
    }
}
```

```
***
**
*
```

21. Complete the following recursive function to calculate  $x^n$ ,  $n$  being a non-negative integer.

```
float find_power(float x, int n){

    if(n == _____) /* 1 Mark */

        return _____; /* 1 Mark */
    else
        return _____;} /* 1 Mark */
}
```

```
0
1
x*find_power(x,n-1)
```

22. What will be the output produced by the following C code?

```
#include<stdio.h>
int num=7;
int r();
int main(){
    for(r();r();r())
        printf("%d ",r());
    return 0;}

int r(){
    return num--;}

```

5 2

23. What are printed out by the following C program?

```
#include <stdio.h>
int confusion(int x, int y) {
    x = 2*x + y;
    return x;}
int main(void) {
    int x = 2, y = 5;
    y = confusion(y, x);
    x = confusion(y, x);
    printf("%d %d\n", x, y);
    return 0;}

```

26 12

24. What will be the output produced by the following C code?

```
#include<stdio.h>
int funct();
main() {
    int i;
    i = funct();
    printf("%d", i);}
int funct(){
    int j = 8, k = 10;
    if(j<k) return (j+k);
    printf("%d ", k-j);}

```

18

▷ The answer is just “18”. If one writes any other like “2 18”, then he/she will get 1 mark.



---

## Section C: 4 Mark Questions

---

25. What will be the value of `sum` after the following code segment executes?

```
int sum = 0, i = 3;
while (i < 100){
    sum = sum + i;
    i = i + 3;}
```

$99 \times 17 = 1683$

26. For the following program, determine the number of times `sum` will be printed.

```
#include <stdio.h>
main(){
    float sum = 0.0, j = 1.0, i = 2.0;
    while (i/j > 0.0625){
        j = j + j;
        sum = sum + i/j;
        printf("%f\n", sum);}
}
```

5

▷ Since there is one line of output for each loop, we need to determine the number of times the loop executes. Since  $i$  is constant, we need to see the growth of  $j$  only. Let the successive values of  $j$  be denoted by  $j_1 = 1, j_2 = 2, j_3 = 4, j_4 = 8, \dots$ , which forms the GP:  $j_n = 2j_{n-1}$ . The `while` loop will execute as long as  $i/j = 2/j > 0.0625 = 2^{-4}$ . As  $i/j$  is in GP with first term  $a = i/j_1 = 2$  and common ratio  $r = \frac{1}{2}$ , the 6th term is  $ar^5 = 2 \cdot (\frac{1}{2})^5 = 2^{-4}$ . Hence, the loop will execute 5 times.

27. The following code segment should read a positive integer `n` and generate a number pattern in triangular format. Examples for `n = 1, 2, 3, 4` are given below.

n = 1	n = 2	n = 3	n = 4
1	212	32123	4321234
	2	323	43234
		3	434
			4

For the program to behave as required, write the expressions for `alpha1` and `alpha2`.

```
int n, i, j;
printf("Enter n: ");
scanf("%d", &n);
for (i = 1; i <= n; i++) {
```

```

for (j = 1; j < i; j++) printf(" ");
for (j = n; j >= i; alpha1)   printf("%d", j);
for (j = i + 1; alpha2; j++) printf("%d", j);
printf("\n");}

```

```

alpha1:  j--
alpha2:  j<=n

```

---

### Section D: $7\frac{1}{2}$ Mark Questions

---

28. The following main function reads  $x$  and  $n$ , and invokes the `fact` function to compute and print the sum of the series:  $x - x^2 2! + x^3 3! \dots + (-1)^n x^n n!$ . However, the two `for` statements are incomplete. Complete them on the dashed line provided.

```

double fact(int n) {
    int i; double f = 1;
    for(_____) /* 2.5 Marks */
        f = f * i ;
    return f; }

```

```

main() {
    int i, n; double x, s, t;
    printf("Enter x, n: ");
    scanf("%lf%d", &x, &n);
    s = x, t = -x * x;
    for(_____) /* 5 Marks */
        s += t * fact(2 * i);
    printf("The sum of series is %f.\n", s);} /* end main */

```

```

for(i = 1; i <=n/2 ; i++) /* 2.5 Marks */
for(i = 2; i <= n; i++, t *= -x) /* 5 Marks */
or
for(i = n/2; i >=1 ; i--) /* 2.5 Marks */
for (i = 2; i <= n; i++, t *= -x) /* 5 Marks */
or
for(i = 1; i <=n ; i++) /* 2.5 Marks */
for (i = 2; i <= n; i++, t *= -x*fact(i)/fact(2*i)) /* 5 Marks */

```

29. The following C-function named `ParkingFee` takes the following two parameters:
- (i) Type of the vehicle, (`vehicleType` = 'M' for motorbike, 'C' for car, and 'B' for Bus);
  - (ii) Number of hours (`parkHour`) that a vehicle has spent in the parking lot.

The function should return the parking charge based on the following parking rates: Rs. 5, Rs. 10 and Rs. 50 per hour respectively for motorbike, car and bus.

Please fill up the parts indicated by dashed lines to make it work correctly. Each dashed line should be filled with one or more C statements.

```

int ParkingFee(char vehicleType, int parkHour){
    int parkFee;

    switch(vehicleType){

        case 'M': -----

        case 'C': -----

        case 'B': -----

        default : -----
    }
    return parkFee;}

```

```

case 'M': parkFee = 5 * parkHour; break;
case 'C': parkFee = 10 * parkHour; break;
case 'B': parkFee = 50 * parkHour; break;
default : parkFee = 0; break;

```

30. When the following C-function **fround** receives a positive floating-point number **x** as a parameter, it should return the rounded value of **x** to two decimal places. For example, **fround**(248.765012) = 248.77, **fround**(248.765000) = 248.76, **fround**(248.764805) = 248.76. In the code for **fround** shown below, the parts indicated by dashed line have been left out for you to fill up.

```

float fround(float x){
    float y, fraction;
    int wholeNumber;

    y = -----;

    wholeNumber=(int)y;
    fraction=y-(float)wholeNumber;

    if (-----)

        wholeNumber=wholeNumber+1;

    return -----;
}

```

```

x*100.0
fraction > 0.5
(float)(wholeNumber/100.0)

```

31. Consider the following power series:

$$\cosine(x) = 1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \frac{x^6}{6!} + \dots$$

To evaluate this series up to a certain level of accuracy, the following function named `cosine` should take a floating-point variable `x` and compute an approximate value of `sum` by adding the successive `terms` inside the `while` loop until the `term` value becomes less than 0.0005. It should then return the value of `sum` as a `double` data type. Fill up the missing parts so that the function can do the required computation.

```
double cosine(float x){
    int i;
    double term,sum;
    term=1; i=0; sum=1;

    while(term>=0.0005) {

        term = -----

        sum = -----

        i = -----

    }
    return(sum);}

```

```
term = term * x * x / (double)((i+1)*(i+2));
sum = sum + term;
i = i+2;
or
term = term * x * x / (double)((2i+1)*(2i+2));
sum = sum + term;
i = i+1;

```