

Answer all questions. Write your answers only in the space provided.

Full marks = 50

1. Answer the following:

[2+3+2+1=8 marks]

- a) What are the minimum and maximum numbers that can be represented in 10-bit 2's complement representation?

Minimum: -512

Maximum: +511

- b) What is the binary equivalent of the decimal number 230.625?

11100110 . 101

- c) What are the 2's complement representations of +21 and -11 in 6-bits? Show how you can subtract 11 from 21 using 6-bit 2's complement arithmetic.

+21 : 010101

-11 : 110101

```

      0 1 0 1 0 1
      1 1 0 1 0 1
      ───────────
      1 0 0 1 0 1 0
```

- d) What will the following program segment print?

```
int a, *p;
double b, *q;
if (sizeof(p) == (sizeof(q))) printf ("\nHello");
else printf ("\nGood day");
```

Hello

2. Answer the following.

[3+2+2+2+3=12 marks]

- a) Consider an array containing the following 8 integers:

30 10 70 20 50 60 80 40

Suppose we want to sort the array using selection sort. Show the contents of the array after every pass.

Initial: 30 10 70 20 50 60 80 40

Pass 1: 10 30 70 20 50 60 80 40

Pass 2: 10 20 70 30 50 60 80 40

Pass 3: 10 20 30 70 50 60 80 40

Pass 4: 10 20 30 40 50 60 80 70

Pass 5: 10 20 30 40 50 60 80 70

Pass 6: 10 20 30 40 50 60 80 70

Pass 7: 10 20 30 40 50 60 70 80

- b) It is required to merge two sorted arrays containing m and n elements respectively to form a sorted array of (m+n) elements. What is the minimum and maximum number of possible comparisons required for the purpose in terms of m and n?

Minimum:	$\min(m, n)$
Maximum:	$m+n-1$

- c) What does the following program print?

```
#include <stdio.h>
int main(){
    int i, p[][4]={{1,2,3,4},{5,6,7,8},{9,10,11,12}};
    for (i=0;i<3;i++) printf("\n (%d, %d) \n", *p[i], (*p)[i]);
}
```

(1, 1)
(5, 2)
(9, 3)

- d) What does the following program print?

```
#include <stdio.h>
int main(){
    int i, j, k, l, a[4][4]={{1,2,3,4},{1,2,3,4},{1,2,3,5},{1,2,3,4}},
        flag1=0, flag2=0;
    int b[4][4]={{1,2,3,4},{2,4,6,8},{1,2,3,4},{1,2,3,4}};
    for (i=0;i<3;i++) for (j=i+1;j<4;j++)
        for (k=0;k<3;k++) for (l=k+1;l<4;l++)
        {
            if ((a[i][k]*a[j][l]-a[j][k]*a[i][l])!=0) flag1=1;
            if ((b[i][k]*b[j][l]-b[j][k]*b[i][l])!=0) flag2=1;
        };
    printf("%d :: %d\n", flag1, flag2); }
```

1 :: 0

- e) Consider a partition P containing the following numbers in the context of the quicksort algorithm:

P = (50 20 100 10 80 70 60 65)

Assuming that the first element (that is, 50) is chosen as the pivot element, show the steps involved in one pass of the partitioning process that places the pivot element in its final position in the sorted list.

50	<u>20</u>	100	10	80	70	60	<u>65</u>
50	20	<u>100</u>	10	80	70	60	<u>65</u>
50	20	<u>100</u>	10	80	70	<u>60</u>	65
50	20	<u>100</u>	10	80	<u>70</u>	60	65
50	20	<u>100</u>	10	<u>80</u>	70	60	65
50	20	<u>100</u>	<u>10</u>	80	70	60	65
50	20	<u>10</u>	<u>100</u>	80	70	60	65
<u>10</u>	20	<u>50</u>	100	80	70	60	65

3. Answer the following.

[3+4+3=10 marks]

- a) Study the following function and count the number of times it is called and the value returned when it is invoked as **guess(5)**.

```
int guess (int m)
{
    int temp;
    if (m < 2) return 1;
    else temp = 5*guess(m-1) + 3*guess(m-2);
    return temp;
}
```

Times called: 15

Value returned: 1324

- b) Consider a 2-dimensional array **MAT** with **N** number of rows and variable number of columns in the different rows. The number of columns in row **r** is **2+r*3**. Show a C code segment to dynamically allocate space for the array **MAT** as mentioned, including all relevant variable declarations.

```
int **MAT, r, col;
MAT = (int **) malloc (N * sizeof(int *));
for (r=0; r<N; r++)
{
    col = 2 + r*3;
    MAT[r] = (int *) malloc (col * sizeof(int));
}
```

- c) The array **int a[10][10]** is filled with an $r \times c$ ($0 \leq r, c < 10$) matrix where the elements are initialized as: **a[i][j] = 10*(i%3)+j**. What will be printed due to the function call:

```
int (*xx)[10] = &aa[2][5];
what (xx, 3, 4);
```

where the function definition is as follows.

```
void what(int x[][10], int r, int c){
    int i, j;
    for (i=0; i<r; i++){
        for (j=0; j<c; j++) printf("%4d ", x[i][j]);
        putchar('\n');
    }
}
```

25	26	27	28
5	6	7	8
15	16	17	18

4. Answer the following.

[3+4+3=10 marks]

- a) A stack of integer numbers is implemented as a linked list with the following definition of a node:

```
struct stack {
    int element;
    struct stack *next;
};
typedef struct stack S;
S *mystack;
```

Complete the function below to push an element **N** into the stack.

```
void push ( S ** top, int N )
{
    S *xyz;
    xyz = (S *) malloc (sizeof(S));

    xyz->element = element;

    xyz->next = *top;

    *top = xyz;
}
```

- b) Consider a stack implementation with the following function prototypes:

void push(S,ch); : push a character 'ch' in the stack **S**

int pop(S); : pop a character from the stack **S**, and return it

int isempty(S); : returns 1 if the stack **S** is empty; returns 0 otherwise

Write a function with prototype

void reverse (stack S, char str1[], char str2[]);

that takes a given string **str1**, reverses it, and stores the resulting string in **str2**, using a stack **S**.

You must use the given functions **push**, **pop** and **isempty** for the purpose.

```
void reverse (stack S, char str1[], char str2[])
{
    char ch;
    int i=0;

    while (i<strlen(str1)) {push(S,str1[i]);i=i+1;};
    i=0;

    while(!isempty(S)) {str2[i]=(pop(S)+'0');i=i+1;};

    str2[i]='\0';
}
```

- c) A linked list is created with the following definition of a node:

```
struct list {
    int roll;
    char name[30];
    struct list *next;
};
typedef struct list L;
L *head;
```

Assume that the linked list is populated with the following data in sequence:

<100,Ramesh>, <102,Dinesh>, <105,Bipin>, <110,Kunal>, <111,Kanti>,
<115,Champak>, <117,Shaunak>, <118,Jatin>.

What will the following program print, when called as: **mystery(head)**?

```
void mystery (node *h)
{
    node *abc;
    abc = h;

    for (;;)
    {
        if (abc == NULL) break;
        printf ("\n%d %s", abc->roll, abc->name);
        if (abc->next == NULL) break;
        abc = (abc->next)->next;
    }
}
```

100 Ramesh
105 Bipin
111 Kanti
117 Shaunak

5. Answer the following.

[3+3+3+1=10 marks]

- a) Complete the following function for reversing the elements of a linked list.

```
struct node {
    int data;
    struct node *next;
};

struct node* reverse (struct node *hea )
{
    struct node *middle;
    struct node *trail;
    middle = NULL;

    while (head != NULL)
    {
        trail = middle;
        middle = head;

        head = _____ head -> next _____ ;

        _____ middle -> next _____ = trail;
    }

    return _____ middle _____ ;
}
```

- b) Complete the following program for printing the number of occurrences of the character Q in the input file “**input.txt**”. Output must come to the terminal.

```
#include <stdio.h>
void main( )
{
    FILE *fp, *fopen( );
    char c;
    int count=0;

    fp = fopen( "input.txt" , "r" );

    while ((c = fgetc(fp) ) != EOF)
        if (c == 'Q') count++;

    fprintf( stdout , "count = %d", count);
}
```

- c) What will the following program print?

```
int memory[100];
int count;

int fib( int n )
{
    count++;
    if (memory[n-1] < 0) memory[n-1] = fib(n-1);
    if (memory[n-2] < 0) memory[n-2] = fib(n-2);
    return (memory[n-1] + memory[n-2]);
}

int main( )
{
    int k, n;
    memory[0]=0; memory[1]=1; count=0;
    for (k=2; k<100; k++) memory[k] = -1;
    n = fib(6);
    printf("n= %d, count= %d", n, count);
}
```

n= 8, count= 11

- d) What will the following program print if it is executed with the following command?

./a.out Hello World

```
void main( int argc, char *argv[ ] )
{
    printf("%d, %s, %s", argc, argv[0], argv[1]);
}
```

3, ./a.out, Hello