1. Write a program to compute the roots of a quadratic equation $ax^2 + bx + c$=0 use following conditions.

   i. There is only one root, if a =0 (x = -c/b).

   ii. There are no real roots, if b^2-4ac is negative.

   iii. If b^2-4ac is positive there are two roots

SOLUTION:

```
#include<stdio.h>
#include<math.h>
int main()
{
   int a,b,c;
   double x1,x2,x,d;

   printf("Input the co-efficients a,b,c values of equation ax^2+bx+c=0\n");
   scanf("%d %d %d",&a,&b,&c);

   d= b*b-4*a*c;

   if(d>0)
   {
      printf("two values of x\n");
      x1= (-b+sqrt(d))/(2*a);
      x2= (-b-sqrt(d))/(2*a);
      printf("x1= %.2lf\t x2= %.2lf\n",x1,x2);
   }
   else if(a==0)
   {
      printf("One solution\n");
      x= -c/b;
      printf("x= %.2lf",x);
   }
   else
      printf("Imaginary Solution");

   return 0;
}
```

2. Given a mark ranging from 0 to 100, write a program to find the grade using **switch** statement.

```
#include<stdio.h>

int main()
{
 int marks;

 printf("\nEnter The Marks Between 0 To 100:");
 scanf("%d", &marks);

 if(marks>100 || marks<0)
  {
   printf("\nInvalid Mark\n");
  }
  else
  {
     switch(marks/10)
     {
        case 10 :
        case 9 :
            printf("\nYour Grade Is: A+ or Excellent\n");
            break;
        case 8 :
        case 7 :
            printf("\nYour Grade Is: A or Very Good\n" );
            break;
        case 6 :
            printf("\nYour Grade Is: B or Fair\n" );
            break;
        case 5 :
        case 4 :
            printf("\nYour Grade Is: C or Pass\n86");
            break;
        default :
           printf("\nYou Grade Is: F or Fail\n");
     }
   }
  return 0;
}
```

3. Write a program that can say the name of weekdays by receiving 1 to 7. Using
    i. If-else statement, and
    ii. Switch Statement

i. if-else Solution:

```
#include<stdio.h>
int main()
{
   int input;

   printf("The days of week Starts with Saturday. Enter from 1 to 7:\t");
   scanf("%d", &input);

   if(input==1)
      printf("Saturday\n");
   else if(input==2)
      printf("Sunday\n");
   else if(input==3)
      printf("Monday\n");
   else if(input==4)
      printf("Tuesday\n");
   else if(input==5)
      printf("Wednesday\n");
   else if(input==6)
      printf("Thursday\n");
   else if(input==7)
      printf("Friday\n");
   else
      printf("Wrong input\n");

   return 0;
}
```

ii. Switch Statement

```
#include<stdio.h>
int main()
{
    int input;
    printf("The days of week Starts with Saturday.\nEnter from 1 to 7:\t");
    scanf("%d", &input);
    switch(input)
    {
        case 1:
        printf("Saturday\n");
        break;
        case 2:
        printf("Sunday\n");
        break;
        case 3:
        printf("Monday\n");
        break;
        case 4:
        printf("Tuesday\n");
        break;
        case 5:
        printf("Wednesday\n");
        break;
        case 6:
        printf("Thursday\n");
        break;
        case 7:
        printf("Friday\n");
        break;
        default:
        printf("Wrong input\n");
        break;
    }
    return 0;
}
```

4. Write a program to check the Leap Year in C

```
#include <stdio.h>
int main()
{
  int year;

  printf("Enter a year to check if it is a leap year\n");
  scanf("%d", &year);

  if (year%400==0 ||( year%4==0 && year%100!=0) )
    printf("%d is a leap year.\n", year);
  else
    printf("%d isn't a leap year.\n", year);

  return 0;
}
```

5. Write a program that convert temperature to from Fahrenheit to Celsius

```
#include <stdio.h>

int main()
{
   float C,F;

   printf("Enter Temperature in Fahrenheit:\t");
   scanf("%f", &F);

   C=((F-32)*5)/9;
   printf("The temperature in Celsius Scale %.2f\n", C);

   return 0;
}
```

6. Write a program to check vowel and consonant using **switch case** statement.

```
#include <stdio.h>

int main()
{
   char cha;

   printf("\nEnter an Alphabet to check if it's Vowel or Consonant:  ");
   scanf("%c",&cha);

   if(cha<=65 || cha>=122)
     {
      printf("Enter Alphabet not other characters\n");
     }
    else
     {
      switch(cha)
      {
         case 'a':
         case 'e':
         case 'i':
         case 'o':
         case 'u':
         case 'A':
         case 'E':
         case 'I':
         case 'O':
         case 'U':
            printf("%c is a Vowel\n",cha);
            break;
         default:
            printf("%c is a Constant\n",cha);
            break;
         }
      }
   return 0;
}
```

7. Swapping between two numbers using a temporary variable

```
#include<stdio.h>

int main()
{
   int a, b, temp;

   printf("Enter value of a:\t");
   scanf("%d",&a);

   printf("Enter value of b:\t");
   scanf("%d",&b);

   printf("Before Swapping:\n");
   printf(" a = %d\n",a);
   printf(" b = %d\n",b);

   temp=a;
   a=b;
   b=temp;

   printf("After Swapping:\n");
   printf(" a = %d\n",a);
   printf(" b = %d\n",b);

   return 0;
}
```

8. Write C program to find Factorial

```
#include <stdio.h>

int main(){

  int c, n, fact = 1;
  printf("Enter a number to calculate its factorial\n");

  scanf("%d", &n);
  for (c = 1; c <= n; c++)
       fact = fact * c;
```

```
    printf("Factorial of %d = %d\n", n, fact);
   return 0;
  }
```

9. Write C program to find Factorial of a number using **recursion**.

```
#include<stdio.h>

int fibo(int);

int main()
{
   int n,i;

   printf("Enter the limit:\n");
   scanf("%d",&n);

   printf("The Fibonacci Series\n");

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

   return 0;
}
int fibo(int x)
{
   if(x==0 ||x==1)
      return x;
   else
      return(fibo(x-2)+fibo(x-1));
}
```

11.Given a number, write a program using while loop to reverse the digits of the number

```
#include<stdio.h>

int main()
{
    int num, rev;

    printf("Enter a number to reverse it:\t");
    scanf("%d", &num);

    rev=0;

    while(num !=0)
    {
       rev=rev*10;
       rev=rev + num%10;
       num=num/10;
    }

    printf("\n Reversed value= %d\n", rev);
    return 0;
 }
```

12. Palindrome Number Program in C using string functions

```
#include <stdio.h>
#include <string.h>

int main()
{
   char a[100], b[100];

   printf("Enter a string to check if it is a palindrome\n");
   gets(a);

   strcpy(b, a); // Copying input string
   strrev(b); // Reversing the string

   if (strcmp(a, b) == 0) // Comparing input string with the reverse string
   printf("The string is a palindrome.\n");
   else
   printf("The string isn't a palindrome.\n");
   return 0;
}
```

13. C program for palindrome without using string functions

```
#include <stdio.h>

int main()
{
   int n, r = 0, t;

   printf("Enter an integer to check if it is palindrome or not\n");
   scanf("%d", &n);

   t = n;

   while (t != 0)
   {
   r = r * 10;
   r = r + t%10;
   t = t/10;
   }

   if (n == r)
   printf("%d is a palindrome number.\n", n);
   else
   printf("%d isn't a palindrome number.\n", n);

   return 0;
}
```

14. Program to Calculate 1 + 2 + 3 + 4 + 5 + ... + n series (sum of any types of series)

```
#include<stdio.h>

int main()
{
   int i,n,sum=0;
```

```
    printf("Enter limit:  ");
    scanf("%d",&n);

    for(i=1;i<=n;i++)
       sum=sum+i;

    printf("\nSum from 1 up to %d is %d\n", n, sum);
    return 0;
}
```

15.Write a program that will read a positive integer determine and print its binary equivalent

```
#include <stdio.h>

int binary_conversion(int);

int main()
{
   int num, bin;

   printf("Enter a decimal number: ");
   scanf("%d", &num);
   bin = binary_conversion(num);
   printf("The binary equivalent of %d is %d\n", num, bin);
}

int binary_conversion(int num)
{
    if (num == 0)
    {
        return 0;
    }
    else
    {
        return (num % 2) + 10 * binary_conversion(num / 2);
    }
}
```

16. Write a program that finds maximum and minimum between two numbers. If the number is equal it gives a message “Equal”.

```
#include<stdio.h>

int main()
{
   long int a,b;

   printf("Enter two numbers which u want to compare\n");
   scanf("%d %d",&a,&b);

   if(a>b)
      printf("\n%d is Larger\n",a);
   else if(a<b)
         printf("\n%d is Larger\n",b);
   else
      printf("\nBoth are Equal\n");
   return 0;
}
```

17.Write a program that takes a number from keyboard and finds whether the number is positive, negative or zero.

```
#include<stdio.h>
int main()
{
   int input;

   printf("Enter a number to check\t");
   scanf("%d",&input);

   if(input>0)
      printf("Positive number\n");
   else if(input<0)
      printf("Negative number\n");
   else
      printf("Zero\n");
   return 0;
}
```

18. Write a program that takes a number from keyboard and prints ‘Y’ if the number is greater than or equal 30 and not equal to 50. Otherwise it prints ‘No’.

```
#include<stdio.h>
int main()
{
    int in;

    printf("Enter a number:  ");
    scanf("%d",&in);

    if(in>=30 && in!=50)
        printf("Y\n");
    else
        printf("N\n");

    return 0;
}
```

19. Write a program to check EVEN and ODD number using **array**.

```
#include<stdio.h>
int main()
{
    int i,array[100],N;

    printf("Enter Size of array:\n");
    scanf("%d",&N);
    printf("\nEnter array elements:\n");

    for(i=0;i<N;i++)
        scanf("%d",&array[i]);


    for(i=0;i<N;i++)
    {
            if(array[i]%2==0)
               printf("%d Even\n",array[i]);
            else
               printf("%d Odd \n",array[i]);
         }

         return 0;
      }
```

20.Write a program to check Prime and Non-prime number using **array**

```
#include <stdio.h>
int main()
{
    int a[100],n,i,j,c=0;
    printf("enter the number= ");
    scanf("%d",&n);
    printf("enter the elements = \n");
    for(i=0; i<n; i++)
        {
        scanf("%d",&a[i]);
        }
    for(i=0; i<n; i++)
    {
        for(j=1; j<=a[i]; j++)
          {
           if(a[i]%j==0)
               c++;
          }
        if(c==2)
        printf("%d is prime\n",a[i]);
        else
        printf("%d is non-prime\n",a[i]);

        c=0;
    }
    return 0;
}
```

21. Print numbers from 1 to 10 using goto statement without any loop

```
#include<stdio.h>
int main()
{
   int number=1;

   repeat:
      printf("%d\n", number);
      number++;

   if(number<=10)
      goto repeat;

   return 0;
}
```

22. Write a C program to multiply/addition two **matrixes**

```
#include <stdio.h>
int main()
{
   int a[10][10], b[10][10], mult[10][10], r1, c1, r2, c2, i, j, k;
   printf("Enter rows and column for first matrix:\n ");
   scanf("%d %d", &r1, &c1);
   printf("Enter rows and column for second matrix:\n ");
   scanf("%d %d",&r2, &c2);

   while (c1!=r2)
   {
   printf(" Error! column of first matrix not equal to row of second.\n");
   printf("Enter rows and column for first matrix: ");
   scanf("%d %d", &r1, &c1);
   printf("Enter rows and column for second matrix: ");
   scanf("%d %d",&r2, &c2);
   }
   printf("\nEnter elements of matrix 1:\n");
   for(i=0; i<r1; ++i)
   {
      for(j=0; j<c1; ++j)
      {
      printf("Enter elements a%d%d: ",i+1,j+1);
```

```
        scanf("%d", &a[i][j]);
        }
    }

    printf("\nEnter elements of matrix 2:\n");
    for(i=0; i<r2; ++i)
    {   for(j=0; j<c2; ++j)
        {
        printf("Enter elements b%d%d: ",i+1,j+1);
        scanf("%d", &b[i][j]);
        }
    }

   for(i=0; i<r1; ++i)
    {
        for(j=0; j<c2; ++j)
        {
            mult[i][j]=0;
            for(k=0; k<c1; ++k)
            {
            mult[i][j]+=a[i][k]*b[k][j];
            }
        }
    }

    printf("\nOutput Matrix:\n");
    for(i=0; i<r1; ++i)
    {
        for(j=0; j<c2; ++j)
            printf("%d  ",mult[i][j]);

        printf("\n");
    }
    return 0;
}
```

23. Write a C program to Insert a new element in a specific position in an array where position & new element given through the keyboard

```
#include <stdio.h>

int main()
{
   int array[100], position, c, n, value;
```

```
  printf("Enter number of elements in array\n");
  scanf("%d", &n);

  printf("Enter %d elements\n", n);

  for (c = 0; c < n; c++)
    scanf("%d", &array[c]);

  printf("Enter the location where you wish to insert an element\n");
  scanf("%d", &position);

  printf("Enter the value to insert\n");
  scanf("%d", &value);

  for (c = n - 1; c >= position - 1; c--)
    array[c+1] = array[c];

  array[position-1] = value;

  printf("Resultant array is\n");

  for (c = 0; c <= n; c++)
    printf("%d\n", array[c]);

  return 0;
}
```

24. Write a C program to sort an array by using **bubble sort** method.

```
#include <stdio.h>

int main()
{
 int array[100], n, c, d, swap;

   printf("Enter number of elements\n");
   scanf("%d", &n);
printf("Enter %d integers\n", n);

for (c = 0; c < n; c++)
  scanf("%d", &array[c]);

for (c = 0 ; c < n - 1; c++)
{
  for (d = 0 ; d < n - c - 1; d++)
    {
    if (array[d] > array[d+1]) /* For decreasing order use < */
    {
    swap = array[d];
    array[d] = array[d+1];
    array[d+1] = swap;
    }
  }
}

printf("Sorted list in ascending order:\n");

for (c = 0; c < n; c++)
printf("%d\n", array[c]);

return 0;
}
```

25. Search a specific number of an array element and say its position if it is found using binary search

```
#include <stdio.h>
int binary_search();
int sort(int b[],int n);

int a[50], n, item, loc, beg, mid, end, i;
int main()
{
    printf("\nEnter number of elements of array: ");
    scanf("%d", &n);
    printf("\nEnter elements of an array :\n");
    for(i=0; i<n; i++)
        scanf("%d", &a[i]);
    printf("\n Array sorted form:\n");

    sort(a,n);
    for(i=0; i<n; i++)
        printf("%d\n", a[i]);
    printf("\nEnter ITEM to be searched: ");
    scanf("%d", &item);
    binary_search();
    return 0;
}
int sort(int b[],int n)
{
    int i,j,x,temp;
    x=n;
    for(i=0;i<n-1;i++)
    {
        for(j=0;j<n-i-1;j++)
        {
            if(b[j]>b[j+1])
            {
                temp=b[j];
                b[j]=b[j+1];
               b[j+1]=temp;

            }
        }
    }
}
int binary_search()
{
    beg = 0;
    end = n-1;
    mid = (beg + end) / 2;
    while ((beg<=end) && (a[mid]!=item))
    {
        if (item < a[mid])
            end = mid - 1;
        else
            beg = mid + 1;
        mid = (beg + end) / 2;
    }
    if (a[mid] == item)
        printf("\n\nITEM found at location %d", mid+1);
    else
        printf("\n\nITEM doesn't exist");
}
```

26. Implementation of Stack Using Array in C

```
#include<stdio.h>

int stack[100],choice,n,top,x,i;
void push(void);
void pop(void);
void display(void);
int main()
{

    top=-1;
    printf("\n Enter the size of STACK[MAX=100]:");
    scanf("%d",&n);
    printf("\n\t STACK OPERATIONS USING ARRAY");
    printf("\n\t--------------------------------");
    printf("\n\t 1.PUSH\n\t 2.POP\n\t 3.DISPLAY\n\t 4.EXIT");
    do
    {
        printf("\n Enter the Choice:");
        scanf("%d",&choice);
        switch(choice)
        {
            case 1:
            {
                push();
                break;
            }
            case 2:
            {
                pop();
                break;
            }
            case 3:
            {
                display();
                break;
            }
            case 4:
            {
                printf("\n\t EXIT POINT ");
                break;
            }
            default:
            {
                printf ("\n\t Please Enter a Valid Choice(1/2/3/4)");
            }

        }
    }
    while(choice!=4);
    return 0;
}
void push()
{
    if(top>=n-1)
    {
        printf("\n\tSTACK is over flow");
    }
    else
    {
        printf(" Enter a value to be pushed:");
        scanf("%d",&x);
        top++;
        stack[top]=x;
    }
}
void pop()
{
    if(top<=-1)
    {
        printf("\n\t Stack is under flow");
    }
    else
    {
        printf("\n\t The popped elements is %d",stack[top]);
        top--;
    }
}
void display()
{
    if(top>=0)
    {
        printf("\n The elements in STACK \n");
        for(i=top; i>=0; i--)
            printf("\n%d",stack[i]);
        printf("\n Press Next Choice");
    }
    else
    {
        printf("\n The STACK is empty");
    }

}
```

27. Copy string in C without using strcpy () function

```
#include <stdio.h>

int main()
{
 int c = 0;
 char s[1000], d[1000] = "What can I say about my programming skills?";

    printf("Before copying, the string: %s\n", d);

    printf("Input a string to copy\n");
    gets(s);

    while (s[c] != '\0')
    {
        d[c] = s[c];
        c++;
    }

    d[c] = '\0';

    printf("After copying, the string: %s\n", d);

    return 0;
}
```

28. Given a string from the keyboard. Write a program to calculate the length of a string

```
#include <stdio.h>
#include <string.h>
int main()
{
     char str[100];

     printf("Enter a string:\n");
```

```
    str[]=getch();

    printf("Length of your name: %d", strlen(str));
    return 0;
}
```

29. C Program to Count All Occurrence of a Character in a String.

```
#include<stdio.h>
#include<string.h>
int main()
{
   char string[100],cha;
   int i,count=0;

   printf("Enter your string:\t ");
   gets(string);

   printf("Enter character you want to search: ");
   cha=getchar();

   for(i=0;i<strlen(string);i++)
   {
      if(string[i]==cha)
      {
         count++;
      }
   }

   printf("%c occurs %d times in the given string\n",cha,count);
   return 0;
}
```

30. C Program To Count the Occurrence of a Substring in String

```
#include <stdio.h>
#include <string.h>

char str[100], sub[100];
int count = 0, count1 = 0;

int main()
{
  int i, j, l, l1, l2;

  printf("\nEnter a string : ");
  gets(str);

  l1 = strlen(str);

  printf("\nEnter a substring : ");
  gets(sub);

  l2 = strlen(sub);

  for (i = 0; i < l1;)
  {
    j = 0;
    count = 0;
    while ((str[i] == sub[j]))
    {
      count++;
      i++;
      j++;
    }
    if (count == l2)
    {
      count1++;
      count = 0;
    }
    else
    i++;
  }
```

```
   printf("%s occurs %d times in %s", sub, count1, str);

   return 0;
}
```

31. Given two integer numbers are we have to swap their values using pointers in C language.

```
#include <stdio.h>
 // function : swap two numbers using pointers
void swap(int *a,int *b)
{
   int t;
    t   = *a;
   *a   = *b;
   *b   =  t;
}

int main()
{
   int num1,num2;

   printf("Enter value of num1: ");
   scanf("%d",&num1);
   printf("Enter value of num2: ");
   scanf("%d",&num2);

   //print values before swapping
   printf("Before Swapping: num1=%d, num2=%d\n",num1,num2);

   //call function by passing addresses of num1 and num2
   swap(&num1,&num2);

   //print values after swapping
   printf("After  Swapping: num1=%d, num2=%d\n",num1,num2);

   return 0;
}
```

32. Store Student Information and Display it Using Structure

```
#include<stdio.h>
struct student {
    char name[50];
    int roll;
    float marks;
     } s;
int main()
{
printf("Enter information:\n");

printf("Enter name: ");
 scanf("%s", s.name);

printf("Enter roll number: ");
scanf("%d", &s.roll);

printf("Enter marks: ");
scanf("%f", &s.marks);

printf("Displaying Information:\n");
printf("Name: ");
puts(s.name);
printf("Roll number: %d\n",s.roll);
printf("Marks: %.1f\n", s.marks);
return 0;
 }
```

33. Find Largest Element Using Dynamic Memory Allocation - calloc()

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
   int i, num;
   float *data;

   printf("Enter total number of elements(1 to 100): ");
   scanf("%d", &num);
```

```
    // Allocates the memory for 'num' elements.
    data = (float*) calloc(num, sizeof(float));

    if(data == NULL)
    {
        printf("Error!!! memory not allocated.");
        exit(0);
    }

    printf("\n");

    // Stores the number entered by the user.
    for(i = 0; i < num; ++i)
    {
       printf("Enter Number %d: ", i + 1);
       scanf("%f", data + i);
    }

    // Loop to store largest number at address data
    for(i = 1; i < num; ++i)
    {
       // Change < to > if you want to find the smallest number
       if(*data < *(data + i))
           *data = *(data + i);
    }

    printf("Largest element = %.2f", *data);

    return 0;
}
```

34. Demonstrate the Dynamic Memory Allocation for Structure

```
 #include <stdio.h>
#include <string.h>
#include <stdlib.h>

int main()
{
    char *mem_allocation;
    /* memory is allocated dynamically */
    mem_allocation = malloc( 20 * sizeof(char) );
    if( mem_allocation== NULL )
    {
      printf("Couldn't able to allocate requested memory\n");
    }
    else
    {
      strcpy( mem_allocation,"Allocation Approved\n");
    }
    printf("Dynamically allocated memory content : " \
         "%s\n", mem_allocation );
    free(mem_allocation);
}
```

35. A Simple C Program to open, read and close the file

```
#include <stdio.h>
int main()
{
   char ch;
   /* Pointer for both the file*/
   FILE *fpr, *fpw;
   /* Opening file FILE1.C in “r” mode for reading */
   fpr = fopen("C:\\file1.txt", "r");

   /* Ensure FILE1.C opened successfully*/
   if (fpr == NULL)
   {
         puts("Input file cannot be opened");
    }

    /* Opening file FILE2.C in “w” mode for writing*/
    fpw= fopen("C:\\file2.txt", "w");

    /* Ensure FILE2.C opened successfully*/
    if (fpw == NULL)
    {
       puts("Output file cannot be opened");
    }

    /*Read & Write Logic*/
    while(1)
    {
        ch = fgetc(fpr);
        if (ch==EOF)
            break;
        else
            fputc(ch, fpw);
    }

    /* Closing both the files */
    fclose(fpr);
    fclose(fpw);

    return 0;
}
```

Collected and presented by Teresa Jency Bala
ID:19 385 20 113
Department: Computer Science and Engineering(CSE)
Imperial College of Engineering, Khulna