

Decision Making and Branching

Chapter_5

5.1 Write a program to determine whether a given number is “odd” or “even” and print

the message

NUMBER IS EVEN

OR

NUMBER IS ODD

(a) Without using the else option.

(b) With else option.

Algorithm:–

Without using the else option

Step 1: Read x.

Step 2: Check $x \% 2 == 0$.

Step 3: If true then go to step 4 and otherwise go to step 5.

Step 4: Display “The number is even” and exit.

Step 5: Display “The number is odd”.

With else option

Step 1: Read x.

Step 2: Check $x \% 2 == 0$.

Step 3: If true then go to step 4 and otherwise go to step 5.

Step 4: Display “The number is even”.

Step 5: Display “The number is odd”.

-

Program:—

Without using the else option

//Write a program to determine whether a given number is “odd” or “even” and print the message

//NUMBER IS EVEN

//Or

//NUMBER IS ODD

//(a) Without using the else option.

//(b) With else option.

// Date : 13/03/2010

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

```
#include<conio.h>
```

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

```
void main()
```

```
{
```

```
int x;

clrscr();

printf("Enter an integer number: ");

scanf("%d",&x);

if(x%2==0)

{

    printf("The number entered is even");

    getch();

    exit(0);

}

printf("The number entered is odd");

getch();

}
```

Output:—

Enter an integer number: 5

The number entered is odd

With else option

//Write a program to determine whether a given number is “odd” or “even” and print the message

//NUMBER IS EVEN

//Or

//NUMBER IS ODD

//(a) Without using the else option.

//(b) With else option.

// Date: March 13,2010

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

```
#include<conio.h>
```

```
void main()
```

```
{
```

```
    int x;
```

```
    clrscr();
```

```
    printf("Enter an integer number: ");
```

```
    scanf("%d",&x);
```

```
    if(x%2==0)
```

```
        printf("The number entered is even");
```

```
    else
```

```
        printf("The number entered is odd");
```

```
    getch();
```

```
}
```

Output:—

Enter an integer number: 5

The number entered is odd

5.2 Write a program to find the number of and sum of all integers greater than 100 and less than 200 that are divisible by 7.

Algorithm:–

-

Step 1: Store 100 to Num & 0 to Sum.

Step 2: if $\text{Num} \% 7 = 0$ then go to Step 3

Step 3: Compute $\text{Count} = \text{Count} + 1$ & $\text{Sum} = \text{Sum} + \text{Num}$ & $\text{Num} = \text{Num} + 1$.

Step 4: if $\text{Num} \leq 200$ then go to Step 2 otherwise go to Step 5.

Step 5: Display Count & Sum.

-

Program:–

```
//Write a program to find the number of and sum of all
```

```
//integers greater than 100 and less than 200 that are divisible by 7.
```

```
// Date : 13/03/2010
```

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

```
#include<conio.h>
```

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

```

void main()
{
    int Num,Sum,Count;

    clrscr();

    Num=100;

    Sum=Count=0;

    Loop:

    if (Num%i==0)
    {
        Sum=Sum+Num;

        Count=Count+1;
    }

    Num=Num+1;

    if(Num<=100)

        goto Loop;

    printf("Count:- %d\n",Count);

    printf("Sum:- %d",Sum);

}
-

```

5.3 A set of two linear equation two unknowns x_1 and x_2 is given below:

$$ax_1 + bx_2 = m$$

$$cx_1 + dx_2 = n$$

The set has a unique solution

$$x_1 = (md - bn) / (ad - cb)$$

$$x_2 = (na - mc) / (ad - cb)$$

-

Algorithm:—

Step 1: Read a,b,c,d,m and n.

Step 2: Compute $a*d - c*b$ and store the result Dr.

Step 3: Check if $Dr \neq 0$.

Step 4: If true then go to Step 5 and otherwise go to step 9.

Step 5: Compute $(m*d - b*n) / (a*d - c*b)$ and store the result x_1 .

Step 6: Compute $(n*a - m*c) / (a*d - c*b)$ and store the result x_2 .

Step 7: Display x_1 .

Step 8: Display x_2 and go to step 10.

Step 9: Display “The division is not possible”.

Step 10: Stop.

-

Program:—

-

//A set of two linear equation two unknowns x1 and x2 is given below:

// $ax1 + bx2 = m$

// $cx1 + dx2 = n$

// The set has a unique solution

// $x1=(md-bn)/(ad-cb)$

// $x2=(na-mc)/(ad-cb)$

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

```
#include<conio.h>
```

```
void main()
```

```
{
```

```
int a,b,c,d,m,n,Dr;
```

```
float x1,x2;
```

```
clrscr();
```

```
printf("Enter the value of a, b, c, d, m, n: ");
```

```
scanf("%d%d%d%d%d%d",&a,&b,&c,&d,&m,&n);
```

```
Dr=(a*d-c*b);
```

```
if(Dr!=0)
```

```
{
```

```
    x1=(m*d-b*n)/dr;
```

```
    x2=(n*a-m*c)/dr;
```



```

        printf("\n The value of x1= %f \n The value of x2= %f",x1,x2);
    }
    else
        printf("The division is not possible and result is an abrupt value ");
    getch();
}

```

5.4 Given the list of marks ranging from 0 to 100, write a program to compute and print the number of students:

- a) who have obtained more than 80 marks.
- b) who have obtained more than 60 marks
- c) who have obtained more than 40 marks
- d) who have obtained 40 or less marks
- e) in the range 81 to 100
- f) in the range 61 to 80
- g) in the range 41 to 60
- h) in the range 0 to 40

The program should use minimum number of if statements.

5.5 Admission to a professional course is subject to the following conditions:

- a) Marks in mathematics ≥ 60

b) Marks in Physics ≥ 50

c) Marks in Chemistry ≥ 40

d) Total in all three subjects ≥ 200

or

Total in mathematics and physics ≥ 150 .

Given the marks in the three subjects, write a program to process the applications to the eligible candidates.

-

Algorithm:—

-

Step 1: Read Maths, Phy and Chem.

Step 2: Compute Maths+Phy+Chem and store the result in Total

Step 3: Compute Maths+Phy and store the result Total_MP

Step 4: Check Maths ≥ 60 && Phy ≥ 50 && Chem ≥ 40 && Total ≥ 200

Step 5: If Step 4 true then go to step 6 otherwise go to step 7.

Step 6: Display “The candidate is eligible for the course” and go to step 11.

Step 7: Check Total_MP ≥ 150

Step 8: If Step 7 true then go to step 9 otherwise go to step 10.

Step 9: Display “The candidate is eligible for the course” and go to step 11

Step 10: Display “The candidate is not eligible for the course” and go to step 11.

Step 11: Stop.

Program:–

-

//Admission to a professional course in subject to the following conditions:

//a) Marks in mathematics ≥ 60

//b) Marks in Physics ≥ 50

//c) Marks in Chemistry ≥ 40

//d) Total in all three subjects ≥ 200

//or

//Total in mathematics and physics ≥ 150 .

//Given the marks in the three subjects, write a program to process the applications to the eligible candidates.

//Date: 13/03/2010

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

```
#include<conio.h>
```

```
void main()
```

```
{
```

```
    int Maths,Phy,Chem,Total,Total_MP;
```

```
    clrscr();
```

```
    printf("Enter the marks of maths :");
```

```
    scanf("%d",&Maths);
```

```
    printf("Enter the marks of phy :");
```

```

scanf("%d",&Phy);

printf("Enter the marks of chem :");

scanf("%d",&Chem);

Total=Maths+Phy+Chem;

Total_MP=Phy+Maths;

if (Maths>=60 && Phy>=50 && Chem>=40 && Total>=200)

    printf("The candidate is eligible for the admission");

else

{

    if(Total_MP>=150)

        printf("The candidate is eligible for the admission");

    else

        printf("The candidate is not eligible for the admission");

}

getch();

}

```

5.6 Write a program to print a two-dimensional Square Root Table as shown below, to provide the square root of any number from 0 to 9.9

Square Root table

Number 0.0 0.1 0.2.....0.9

0.0

1.0

3.0 x y

...

9.0

-

Algorithm:-

rogram:-

-

//Write a program to print a two-dimensional Square Root Table as shown below,

// to provide the square root of any number from 0 to 9.9

// Square Root table

//Number 0.0 0.1 0.2.....0.9

//0.0

//1.0

//3.0 x y

//...

```
//9.0
```

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

```
#include<conio.h>
```

```
#include<math.h>
```

```
void main()
```

```
{
```

```
    float sq,sum,i,j;
```

```
    clrscr();
```

```
    printf("Number ");
```

```
    j=0.1;
```

```
    loop3:
```

```
        printf("    %f",j);
```

```
        j=j+0.1;
```

```
        if(j<0.5)
```

```
            goto loop3;
```

```
    printf("\n");
```

```
    i=1;
```

```
    loop1:
```

```
        printf("%f",i);
```

```
        j=0.1;
```

```
    loop:
```

```

sum=i+j;

sq=sqrt(sum);

printf("  %f",sq);

j=j+0.1;

if(j<0.5)

    goto loop;

i=i+1;

if(i<=4)

{

    printf("\n");

    goto loop1;

}

getch();

}

```

Output:–

5.9 Write a program that will read the value of x and evaluate the following function

Y= 1 for $x > 0$

0 for $x = 0$

-1 for $x < 0$

Using

- (a) Nested if statements
- (b) Else if statements
- (c) Conditional operators

-

Algorithm

-

-

Step 1: Read x.

Step 2: Check $x > 0$, if true then go to step 3 otherwise go to step 5.

Step 3: Assign 1 to y, and go to step 4

Step 4: Display y and go to step 10.

Step 5: Check if $x == 0$, if true then go to step 6 otherwise go to step 8.

Step 6: Assign 0 to y and go to step 7.

Step 7: Display y and go to step 10.

Step 8: Assign -1 to y, go to step 9.

Step 9: Display y and go to step 10.

Step 10: End

-

Program:—

Else if statements

//Write a program that will read the value of x and evaluate the following function

//Y= 1 for x>0

// 0 for x=0

// -1 for x<0

//Using

//(a) Nested if statements

//(b) Else if statements

//(c) Conditional operators

//Date: 13/03/2010

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

```
#include<conio.h>
```

```
void main()
```

```
{
```

```
    int y;
```

```
    float x;
```

```
    clrscr();
```

```
    printf("Enter the value of X: ");
```

```
    scanf("%f",&x);
```

```
    if(x>0)
```

```

{
    y=1;

    printf("The value of y for the given value of x=%f is %d\n",x,y);

}

else if(x==0)

{

    y=0;

    printf("The value of y for the given value of x=%f is
%d\n",x,y);

}

else

{

    y=-1;

    printf("The value of y for the given value of x=%f is %d\n",x,y);

}

getch();

}

```

Output:–

-

Enter the value of X: 3

The value of y for the given value of x=3 is =1

-

Nested if statements

//Write a program that will read the value of x and evaluate the following function

//Y= 1 for x>0

// 0 for x=0

// -1 for x<0

//Using

//(a) Nested if statements

//(b) Else if statements

//(c) Conditional operators

//Date: 13/03/2010

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

```
#include<conio.h>
```

```
void main()
```

```
{
```

```
    int y;
```

```
    float x;
```

```
    clrscr();
```

```
    printf("Enter the value of X: ");
```

```
    scanf("%f",&x);
```

```
    if(x>0)
```

```
{  
    y=1;  
    printf("The value of y for the given value of x=%f is %d\n",x,y);  
}  
else  
    {  
        if(x==0)  
        {  
            y=0;  
            printf("The value of y for the given value of x=%f is %d\n",x,y);  
        }  
        else  
        {  
            y=-1;  
            printf("The value of y for the given value of x=%f is %d\n",x,y);  
        }  
    }  
    getch();  
}
```

Output:—

Enter the value of X: 3

The value of y for the given value of x=3 is =1

Conditional operators

//Write a program that will read the value of x and evaluate the following function

//Y= 1 for x>0

// 0 for x=0

// -1 for x<0

//Using

//(a) Nested if statements

//(b) Else if statements

//(c) Conditional operators

// Date 13/03/2010

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

```
#include<conio.h>
```

```
void main()
```

```
{
```

```
    int y;
```

```
    float x;
```

```
    clrscr();
```

```
    printf("Enter the value of X: ");
```

```
scanf("%f",&x);
```

```
(x>0?(y=1):(x==0)?(y=0):(y=-1));
```

```
printf("The value of y for the given value of x=%f is %d\n",x,y);
```

```
getch();
```

```
}
```

-

-

Output:-

-

Enter the value of X: 3

The value of y for the given value of x=3 is =1

5.10 Write a program to compute the real roots of a quadratic equation

$$ax^2 + bx + c = 0$$

The roots are given by the equations:

$$X1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

$$X2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

The program should request for the values of the constants a,b and c and print the values of x1 and x2. Use the following rules:

- (a) No solution , if both a and b are zero
- (b) There is only one root,if $a=0$ ($x=-c/b$)
- (c) There is no real root if b^2-4ac is negative
- (d) Otherwise there are real roots.

Test your program with appropriate data.

Algorithm:–

-

Step 1: Read a, b and c

Step 2: Compute $b^2-4*a*c$ and store the result in d.

Step 3: Check if $a==0 \ \&\& \ b==0$, if true then go to step 4 otherwise go to step 5.

Step 4: Display “There is no solution of the quadratic equation” and go to step 13.

Step 5: Check if $a==0$, if true go to step 6 otherwise go to step 8.

Step 6: Compute $x=-c/b$ and go to step 7.

Step 7: Display “There is only one root” and display x and go to step 13.

Step 8: Check if $d < 0$, if true go to step 9 otherwise go to step 10.

Step 9: Display “Roots are imaginary” and go to step 13.

Step 10: Compute $x_1 = (-b + \sqrt{b^2 - 4ac}) / (2a)$ and go to step 11

Step 11: Compute $x_2 = (-b - \sqrt{b^2 - 4ac}) / (2a)$

Step 12: Display x_1 and x_2

Step 13. Stop

Program:—

-

// $ax^2 + bx + c = 0$

// The roots are given by the equations:

// $X_1 = (-b + \sqrt{b^2 - 4ac}) / (2a)$

// $X_2 = (-b - \sqrt{b^2 - 4ac}) / (2a)$

// The program should request for the values of the constants a, b and c and print the values of x_1 and x_2 . Use the following rules:

//(a) No solution, if both a and b are zero

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

//(c) There is no real root if $b^2 - 4ac$ is negative

//(d) Otherwise there are real roots.

// Test your program with appropriate data.

// Date 13 March, 2010


```

#include<stdio.h>

#include<conio.h>

#include<math.h>

void main()

{

    float a,b,c,d;

    float x1,x2,x;

    clrscr();

    printf(" Enter the value of a: ");

    scanf("%f",&a);

    printf("\n Enter the value of b: ");

    scanf("%f",&b);

    printf("\n Enter the value of c: ");

    scanf("%f",&c);

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

    if(a==0 && b==0)

        printf(" There is no solution of the quadratic equation");

    else if(a==0)

    {

        x=-c/b;

        printf(" There is only one root of the equation, x= %f",x);

```

```

    }
else if(d<0)
{
    printf("The roots are imaginary and as follows: \n");
}
else
{
    x1 = (-b+sqrt(d))/(2*a);
    x2 = (-b-sqrt(d))/(2*a);
    printf("The roots are real");
    printf("x1=%f \n x2=%f",x1,x2);
}
getch();
}

```

Output:—

Enter the value of a: 1

Enter the value of b: -3

Enter the value of c: 2

The roots are real

x1=2 x2=1

5.11 Write a program to read three integer values from the keyboard and display the

output stating that they are the sides of right-angled triangle.

Algorithm:–

-

Step 1: Read Len, Hei, Hyp.

Step 2: Compute $\text{Temp1} = \text{Hyp} * \text{Hyp}$, $\text{Temp2} = \text{Len} * \text{Len} + \text{Hei} * \text{Hei}$.

Step 3: Check $\text{Temp1} == \text{Temp2}$ is true then go to Step 4 otherwise go to Step 5

Step 4: Display “Triangle is Right Angle Triangle”.

Step 5: Display “Triangle is Not a Right Angle Triangle”.

-

Program:–

-

```
//Write a program to read three integer values from the keyboard and display the
```

```
//output stating that they are the sides of right-angled triangle.
```

```
//Date : 13/03/2010
```

```
#include<conio.h>
```

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

```
void main()
```

```
{
```

```

float Len,Hei,Hyp;

float Temp1,Temp2;

clrscr();

printf("Enter Length Height and Hypotenes of Triangle\n");

scanf("%f %f %f",&Len,&Hei,&Hyp);

Temp1=Hyp*Hyp;

Temp2=Len*Len+Hei*Hei;

if(Temp1==Temp2)

    printf("Triangle is Right Angle Triangle\n");

else

    printf("Triangle is Not a Right Angle Triangle\n");

getch();

}

```

Output:—

Enter Length Height and Hypotenes of Triangle—

2 3 4

Triangle is Not a Right Angle Triangle

5.12 An electricity board charges the following rates for the use of electricity:

For the first 200 units; 80 P per unit

For the next 100 units; 90 P per unit

Beyond 300 units; Rs. 1 per unit

All users are charged a minimum of Rs. 100 as meter charge. If the total amount is more than Rs. 400, then an additional surcharge of 15% of total amount is charged.

Write a program to read the names of users and number of units consumed and printout the charges with names.

-

Algorithm:—

-

Step 1: Read Name & Units.

Step 2: Check $\text{Units} \geq 0 \ \&\& \ \text{Units} \leq 200$ if true the go to Step 3 otherwise go to Step 4

Step 3: Compute $\text{Charge} = 100 + (\text{Units} * 0.80)$ & go to Step 9

Step 4: Check $\text{Units} > 200 \ \&\& \ \text{Units} \leq 300$ if true the go to Step 5 otherwise go to Step 6

Step 5: Compute $\text{Charge} = 100 + (\text{Units} * 0.90)$ & go to Step 9

Step 6: Check $\text{Units} > 300 \ \&\& \ \text{Units} \leq 400$ if true the go to Step 7 otherwise go to Step 8

Step 7: Compute $\text{Units} > 300 \ \&\& \ \text{Units} \leq 400$ & go to Step 9

Step 8: Compute $\text{Charge} = (100 + \text{units}) + (100 + \text{Units}) * 15$ & go to Step 9

Step 9: Display Name Units Charge

-

Program:—

```
//An electricity board charges the following rates for the use of electricity:

//      For the first 200 units; 80 P per unit

//      For the next 100 units; 90 P per unit

//      Beyond 300 units; Rs. 1 per unit

//All users are charged a minimum of Rs. 100 as meter charge. If the total amount
is more than Rs. 400,

//then an additional surcharge of 15% of total amount is charged.

//Write a program to read the names of users and number of units consumed and
printout the charges with names.

//Date : 13/03/2010

#include<conio.h>

#include<stdio.h>

void main()

{

    int Units;

    char Name[10];

    float Charge;

    clrscr();

    printf("Enter Name of User:-\n");

    scanf("%s",&Name);

    printf("Enetr Total Units Consumed\n");
```

```

scanf("%d",&Units);

if(Units>=0&&Units<=200)

    Charge=100+(Units*0.80);

else if(Units>200&&Units<=300)

    Charge=100+(Units*0.90);

else if(Units>300&&Units<=400)

    Charge=100+Units;

else

    Charge=(100+units)+(100+Units)*15;

printf("Name      Units      Charge\n");

printf("%s      %d      %.2f",Name,Units,Charge);

getch();

}

```

Output:–

Enter Name of User:– Ritesh

Enetr Total Units Consumed 600

Name	Units	Charge
Ritesh	600	805.00

5.13 Write a program to compute and display the sum of all integers that are divisible by 6

but not divisible by 4 and lie between 0 and 100. The program should also count and

display the number of such values.

Algorithm:–

-

Step 1: Store 0 to Sum, Count and i.

Step 2: if $i \% 6 == 0$ & $i \% 4 != 0$ is true then Continue from Step3 otherwise go to Step 5.

Step 3: Display i

Step 4: Compute $\text{Count} = \text{Count} + 1$ & $\text{Sum} = \text{Sum} + 1$.

Step 5: Compute $i = i + 1$

Step 6: if $i \leq 100$ then go to Step 2.

Step 7: Display Sum & Count.

-

Program:–

-

```
//Write a program to compute and display the sum of all integers that are divisible by 6
```

```
// but not divisible by 4 and lie between 0 and 100. The program should also count and
```

```
// display the number of such values.
```

```
//Date : 13/03/2010
```

```
#include<conio.h>
```

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



```
void main()

{

    int Sum,i,Count;

    clrscr();

    Sum=Count=0;

    i=0;

    Loop:

    if((i%6==0)&&(i%4!=0))

    {

        printf(“%d \n”,i);

        Count=Count+1;

        Sum=Sum+i;

    }

    i=i+1;

    if(i<=100)

        goto Loop;

    printf(“Sum of Numbers is %d\n”,Sum);

    printf(“Count of Numbers is %d\n”,Count);

    getch();

}
```

Output:–

6 18 30 42 54 66 78 90

Sum of Numbers is 384

Count of Numbers is 8

5.14 Write an interactive program that could read a positive integer number and decide whether the number is prime number and display the output accordingly. Modify the program to count all the prime numbers that lie between 100 and 200.

Algorithm:–

Step 1: Read Num.

Step 2: Store 2 to i & 0 to Count.

Step 3: Compute Num%i & store the result in Temp.

Step 4: if Temp==0 then Count=Count+1

Step 5: if i<=Num then goto step 3

Step 6: if Count==1 then Display Number is Prime

Step 7: Otherwise Display Number is not Prime.

-

Program:–

```
//Write an interactive program that could read a positive integer number and decide
```

```
//whether the number is prime number and display the output accordingly.
```

```
//Date : 13/03/2010
```

```
#include<conio.h>
```

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

```
void main()

{

    int Num,i,Count,Temp;

    clrscr();

    Count=0;

    i=2;

    printf("Enter A Number :-\n");

    scanf("%d",&Num);

    Loop:

    Temp=Num%i;

    if(Temp==0)

        Cpunt=Count+1;

    i=i+1;

    if(i<=Num)

        goto Loop;

    if(Count==1)

        printf("Number %d is Prime",Num);

    else

        printf("Number %d is Not Prime",Num);

    getch();

}
```

Output:–

Enter A Number :–

6

Number 6 is Prime

5.15 Write a program to read a double-type value x that represent angle in radians and a

character-type variable T that represents the type of trigonometric function and display

the value of

- a) Sin(x), if s or S is assigned to T,
- b) Cos(x), if c or C is assigned to T, and
- c) Tan(x), if t or T is assigned to T.

Using (i) if.....else statement and (ii) switch statement.

Algorithm:–

Step 1: Read x, T.

Step 2: Choice T is s or S then Val=sin(x)

Step 3: Choice T is c or C then Val=cos(x)

Step 4: Choice T is t or T then Val=tan(x)

Step 5: Display Val.

Flowchart:–

Using if.....else

Using switch statement

Program:–

Using if.....else

//Write a program to read a double-type value x that represent angle in radians and a

// character-type variable T that represents the type of trigonometric function and display

// the value of

//a) Sin(x), if s or S is assigned to T,

//b) Cos(x), if c or C is assigned to T, and

//c) Tan(x), if t or T is assigned to T.

//Using (i) if.....else statement

//Date : 13/03/2010

#include<conio.h>

#include<stdio.h>

void main()

{

double x,Val;

char T;

Val=0;

clrscr();

printf("Enter Angle:–\n");

```

scanf("%lf",&x);

printf("\ns or S for Sin(x)");

printf("\nc or C for Cos(x)");

printf("\nt or T for Tan(x)");

printf("\nEnter Choice\n");

T=getch();

if((T=='s')||(T=='S'))

    Val=sin(x);

else if((T=='c')||(T=='C'))

    Val=cos(x);

else if((T=='t')||(T=='T'))

    Val=tan(x);

else

    printf("\nWrong Input\n");

printf("Value:— %lf",Val);

getch();

}

```

Output:—

Enter Angle:—

90

s or S for Sin(x)

c or C for Cos(x)

t or T for Tan(x)

Enter Choice

s

Value:— 1.000000

Using Switch Statement

```
//Write a program to read a double-type value x that represent angle in radians and  
a
```

```
// character-type variable T that represents the type of trigonometric function  
and display
```

```
// the value of
```

```
//a) Sin(x), if s or S is assigned to T,
```

```
//b) Cos(x), if c or C is assigned to T, and
```

```
//c) Tan(x), if t or T is assigned to T.
```

```
//Using (ii) switch statement
```

```
//Date : 13/03/2010
```

```
#include<conio.h>
```

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

```
void main()
```

```
{
```

```
double x,Val;
```

```
char T;
```

```
clrscr();

Val=0;

printf("Enter Angle:—\n");

scanf("%lf",&x);

printf("\ns or S for Sin(x) \ns or S for Cos(x) \ns or S for Tan(x)\nEnter Choice
");

T=getch();

switch(T)

{

    case 's':

    case 'S': Val=sin(x); break;

    case 'c':

    case 'C': Val=cos(x); break;

    case 't':

    case 'T': Val=tan(x); break;

    default:printf("\nWrong Input\n");

}

printf("Value:— %lf\n",Val);

getch();

}
```

Enter Angle:—

90

s or S for Sin(x)

c or C for Cos(x)

t or T for Tan(x)

Enter Choice

s

Value:— 1.000000

Reference:

<http://hstuadmission.blogspot.com/2010/12/solution-programming-in-ansi-c-chapter.html>