

CCP Lab Manual

Prabodh C P

Volunteer

Free Software Movement Karnataka

www.fsmk.org

July 17, 2014

Contents

1	Roots of a Quadratic Equation Program	2
2	Palindrome Check Program	5

Chapter 1

Roots of a Quadratic Equation Program

Design and develop a flowchart or an algorithm that takes three coefficients (a, b, and c) of a Quadratic equation ($ax^2+bx+c=0$) as input and compute all possible roots. Implement a C program for the developed flowchart/algorithm and execute the same to output the possible roots for a given set of coefficients with appropriate messages.

```
/******  
*File : 01Quadratic.c  
*Description : Program to find the roots of a Quadratic Equation  
*Author : Prabodh C P  
*Compiler : gcc compiler 4.6.3, Ubuntu 12.04  
*Date : 16 July 2014  
*****/  
  
#include<stdio.h>  
#include<stdlib.h>  
#include<math.h>  
  
/******  
*Function : main  
*Input parameters: no parameters  
*RETURNS : 0 on success  
*****/  
  
int main(void)  
{  
float fA,fB,fC,fDesc,fX1,fX2,fRealp,fImagp;  
  
printf("\n*****");  
printf("\n*\tPROGRAM TO FIND ROOTS OF A QUADRATIC EQUATION\t\t*\n");  
printf("*****");
```

```

printf("\nEnter the coefficients of a,b,c \n");
scanf("%f%f%f",&fA,&fB,&fC);
if(0 == fA)
{
printf("\nInvalid input, not a quadratic equation - try again\n");
exit(0);
}

/*COMPUTE THE DESCRIMINANT*/
fDesc=fB*fB-4*fA*fC;

if(0 == fDesc)
{
fX1 = fX2 = -fB/(2*fA);

printf("\nRoots are equal and the Roots are \n");
printf("\nRoot1 = %g and Root2 = %g\n",fX1,fX2);
}
else if(fDesc > 0)
{
fX1 = (-fB+sqrt(fDesc))/(2*fA);
fX2 = (-fB-sqrt(fDesc))/(2*fA);
printf("\nThe Roots are Real and distinct, they are \n");
printf("\nRoot1 = %g and Root2 = %g\n",fX1,fX2);
}
else
{
fRealp = -fB / (2*fA);
fImagp = sqrt(fabs(fDesc))/(2*fA);
printf("\nThe Roots are imaginary and they are\n");
printf("\nRoot1 = %g+i%g\n",fRealp,fImagp);
printf("\nRoot2 = %g-i%g\n",fRealp,fImagp);
}

return 0;
}

```

Output

Run the following commands in your terminal:

```
$ gcc 01Quadratic.c -lm
```

```
$ ./a.out
```

```
Enter the coefficients of a,b,c
```

```
1 -4 4
```

```
Roots are equal and the they are
```

```
Root1 = 2.000000 and Root2 = 2.000000
```

```
  $ ./a.out
```

Enter the coefficients of a,b,c

1 -5 6

The Roots are Real and distinct, they are

Root1 = 3.000000 and Root2 = 2.000000

\$/a.out

Enter the coefficients of a,b,c

1 3 3

The Roots are imaginary and they are

Root1 = -1.500000+i0.866025

Root2 = -1.500000-i0.866025

Chapter 2

Palindrome Check Program

Design and develop an algorithm to find the reverse of an integer number NUM and check whether it is PALINDROME or NOT. Implement a C program for the developed algorithm that takes an integer number as input and output the reverse of the same with suitable messages. Ex: Num: 2014, Reverse: 4102, Not a Palindrome

```

/*****
*File : 02Palindrome.c
*Description: Program to check whether the given integer is a Palindrome or not
*Author : Prabodh C P
*Compiler : gcc compiler, Ubuntu 10.04
*Date : 4 July 2012
*****/

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

/*****
*Function : main
*Input parameters: no parameters
*RETURNS : 0 on success
*****/

int main(void)
{
    int iNum,iRev = 0,iTemp,iRem;

    printf("\n*****");
    printf("\n*\tPROGRAM TO CHECK WHETHER AN INTEGER IS A PALINDROME OR NOT\t*\n");
    printf("*****");

    printf("\nEnter a number\n");
    scanf("%d",&iNum);

    iTemp = iNum;
```

```
while(iNum!=0)
{
iRem = iNum % 10;
iRev = iRev * 10 + iRem;
iNum = iNum/10;
}
printf("\nReverse is %d",iRev);

if(iRev == iTemp)
printf("\nNumber %d is a palindrome\n",iTemp);
else
printf("\nNumber %d is not a palindrome\n",iTemp);

return 0;
}
```

Output