**Chapter 7: User-Defined Functions II**

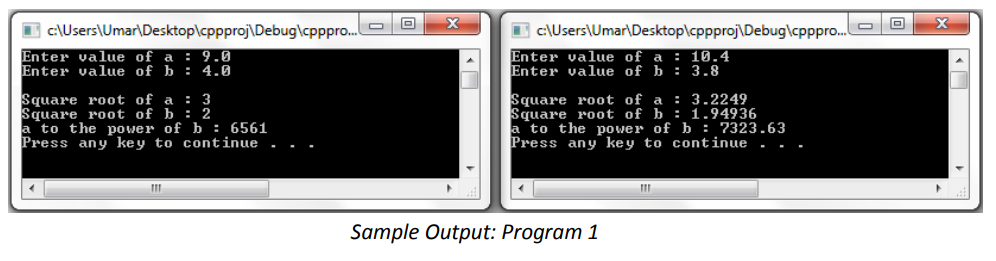
**Laboratory Exercises (2)**

**EXERCISES Solutions**

**Predefined Functions & User defined Functions**

**KEYWORDS: Math.h, function prototype, return, double**

**Program 1:** Write a program that takes as input two real numbers and displays their square root and power as output. You can use the built-in library functions.



#include<iostream>

#include<cmath>

using namespace std;

int main()

{

double a,b;

cout<<"Enter value of a : ";

cin>>a;

cout<<"Enter value of b : ";

cin>>b;

cout<<endl<<"Square root of a : "<<sqrt(a)<<endl;

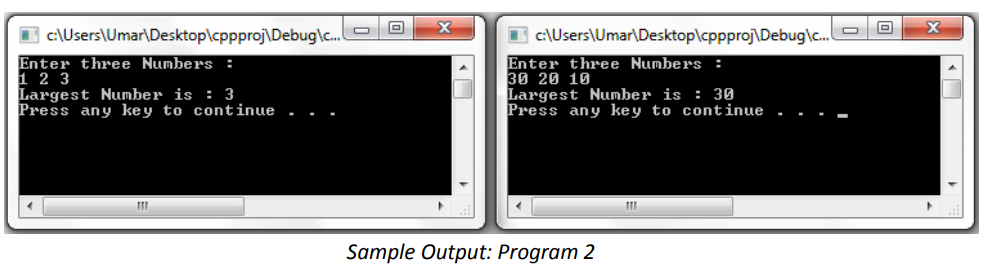
cout<<"Square root of b : "<<sqrt(b)<<endl;

cout<<"a to the power of b : "<<pow(a,b)<<endl;

return 0;

}

**Program 2:** Write a function that takes two integer numbers as input and returns the largest of two number as output. Use the above function to find the largest of three numbers entered by the users.



#include<iostream>

using namespace std;

int largest(int a, int b)

{

if(a > b)

{

return a;

}

else

{

return b;

}

}

int main()

{

int a,b,c;

cout<<"Enter Three Numbers : "<< endl;

cin>>a>>b>>c;

cout<<"Largest Number is : "<<largest(largest(a,b),c)<<endl;

return 0;

}

**Program 3:** Write a value-returning function isVowel that returns the value true if a given character is a vowel and false otherwise. Write a program that prompts the user to input a sequence of characters and outputs the number of vowels. (Use the function isVowel).

Graphical user interface

Description automatically generated

#include<iostream>

using namespace std;

bool isVowel(char a)

{

if((a == 'a') || (a == 'e') || (a == 'i') || (a == 'o') || (a == 'u'))

{

return true;

}

else

{

return false;

}

}

int main()

{

char ch='y'; int count=0;

while(ch=='y')

{

cout<<"Enter value of ch : ";

cin>>ch;

if(isVowel(ch))

count++;

cout<<"Do you want to continue (y/n) ? ";

cin>>ch;

}

cout<<"Number of Vowels = "<<count<<endl;

return 0;

}

**Program 4:** Write a program that takes as input five numbers and outputs the mean (average) and standard deviation of the numbers. If the numbers are x1, x2,x3, x4, and x5, then the mean is x= (x1 + x2 + x3 + x4 + x5)/5 and the standard deviation is:

A picture containing diagram

Description automatically generated

Your program must contain at least the following functions: a function that calculates and returns the mean and a function that calculates the standard deviation.

Graphical user interface, application

Description automatically generated

#include<iostream>

#include<cmath>

using namespace std;

double mean(double x1, double x2, double x3, double x4, double x5)

{

return ((x1+x2+x3+x4+x5)/5);

}

double standarddeviation(double x1, double x2, double x3, double x4, double x5, double x)

{

double nr = ((x1-x)\*(x1-x) + (x2-x)\*(x2-x) + (x3-x)\*(x3-x) + (x4-x)\*(x4-x) + (x5-x)\*(x5-x))/5;

return sqrt(nr);

}

int main()

{

double x1,x2,x3,x4,x5,x;

cout<<"Enter Five Numbers : "<<endl;

cin>>x1>>x2>>x3>>x4>>x5;

x= mean(x1,x2,x3,x4,x5);

cout<<endl;

cout<<"Mean : "<<x<<endl;

cout<<"standard deviation : "<<standarddeviation(x1,x2,x3,x4,x5,x)<<endl<<endl;

return 0;

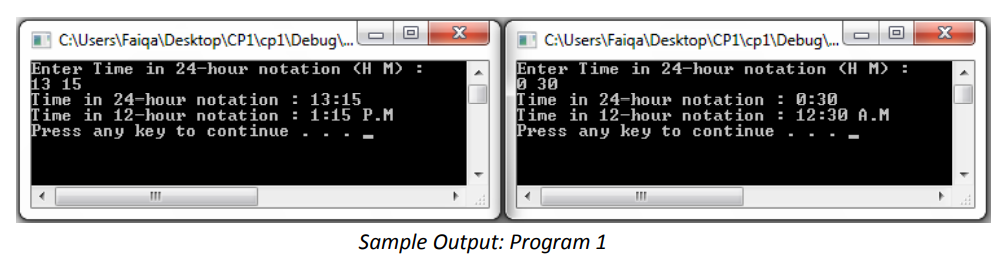
}

**EXERCISES**

**Functions and Parameter Passing**

**KEYWORDS: formal parameter, call by value, call by reference, local variable, global variable, static variable**

**Program 1:** Write a program to convert the time from 24-hour notation to 12-hour notation. Your program must contain a function that converts the time from 24-hour notation to 12-hour notation. For 12-hour time notation, your program must display AM or PM. Use pass by reference method.

****

#include<iostream>

using namespace std;

void convert24to12(int &hours, int minutes, int &ampm)

{

if(hours>12)

{

hours=hours-12;

ampm=1;

}

else

{

if(hours==12)

{

hours=12;

ampm=1;

}

if(hours==0)

{

hours=12;

ampm=0;

}

}

}

int main()

{

int h, m, ampm=0;

cout<<"Enter Time in 24-hour notation (H M) : "<<endl;

cin>>h>>m;

cout<<"Time in 24-hour notation : "<<h<<":"<<m<<endl;

convert24to12(h, m, ampm);

if(ampm==0)

cout<<"Time in 12-hour notation : "<<h<<":"<<m<<" A.M"<<endl;

else

cout<<"Time in 12-hour notation : "<<h<<":"<<m<<" P.M"<<endl;

return 0;

}

**Program 2:** Write a function that accept integer values for day, month and year and return them in valid format. If day is above 30, then it converts day to 30. If month is above 12 then it converts month to 12. Similarly, if day or month is below 1 then it converts them to 1. Finally, it ensures that the years are between 2000 and 2015. If all inputs are valid then it returns true, else it returns false. Write a program that prompts user to input day, month, and year. The program then validates them using the function written above and displays validate if all the input are valid. Otherwise, the program displays invalid and the corrected values of input returned by the function. Use pass by reference method.

**Text

Description automatically generated**

#include<iostream>

using namespace std;

bool validateDate(int &d, int &m, int &y)

{

bool error=true;

if(d > 30)

{ d=30;

error=false;

}

if(d < 1)

{ d=1;

error=false;

}

if(m > 12)

{ m=12;

error=false;

}

if(m < 1)

{

m=1;

error=false;

}

if(y > 2014)

{

y=2014;

error=false;

}

if(y < 2000)

{

y=2000;

error=false;

}

return error;

}

int main()

{

int d,m,y;

cout<<"Enter Date between 1/1/2000 to 30/12/2014 in format (D M Y) : "<<endl;

cin>>d>>m>>y;

cout<<"Date Entered : "<<d<<"/"<<m<<"/"<<y<<endl;

bool isvalid = validateDate(d, m, y);

if(isvalid==true)

{

cout<<"You entered Valid Date : "<<d<<"/"<<m<<"/"<<y<<endl;

}

else

{

cout<<"You entered InValid Date !"<<endl<<"Corrected Date is : "<<d<<"/"<<m<<"/"<<y<<endl;

}

return 0;

}

**Program 3:** Write a function that takes as parameter an integer (as a long value) and returns the number of odd, even, and zero digits. Also write a program to test your function. Use pass by reference method

**Graphical user interface, application

Description automatically generated**

#include<iostream>

using namespace std;

void checkDigits(long int no, int &evens, int &odds, int &zeros)

{

while(no > 0)

{

int digit = no % 10;

if(digit == 0)

{

zeros++;

}

else if(digit % 2 == 0)

{

evens++;

}

else

{

odds++;

}

no=no/10;

}

}

int main()

{

long int no;

int evens=0,odds=0,zeros=0;

cout<<"Enter any no :"<<endl;

cin>>no;

checkDigits(no, evens, odds, zeros);

cout<<"No of Even Digits : "<<evens<<endl;

cout<<"No of Odd Digits : "<<odds<<endl;

cout<<"No of Zero Digits : "<<zeros<<endl;

return 0;

}