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

**Extra Programming Examples**

**Laboratory Exercises (2)**

**Part 1: An Example of Value Parameters**

A Program that illustrates how a value parameter works.

#include<iostream>

using namespace std;

void funcValueParam(int num);

int main()

{

int number = 6;

cout << "Line 2: Before calling the function " << "funcValueParam, number = " << number << endl;

funcValueParam(number);

cout << "Line 4: After calling the function " << "funcValueParam, number = " << number << endl; return 0;

}

void funcValueParam(int num)

{

cout << "Line 5: In the function funcValueParam, " << "before changing, num = " << num << endl;

num = 15;

cout << "Line 7: In the function funcValueParam, "<< "after changing, num = " << num << endl;

}

**Part 2: Examples of Reference Parameters**

**Example 1:** A Program that illustrates how a reference parameter works. This program calculates the factorial of a number.

#include <iostream>

using namespace std;

void getNumber(int &num)

{

cout<<"Enter a number: "<<endl;

cin>>num;

}

void calculateFactorial(int n)

{

int i, fact = 1;

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

{

fact = fact \* i;

}

cout<<"The factorial of "<<n<<" is: "<<fact;

}

int main()

{

int a;

getNumber(a);

calculateFactorial(a);

return 0;

}

**Example 2:** A Program that illustrates how a reference parameter works. This program swaps two numbers using call by reference

#include <iostream>

using namespace std;

void swapNums(int &x, int &y)

{

int z = x;

x = y;

y = z;

}

int main()

{

int firstNum = 10;

int secondNum = 20;

cout << "Before swap: " << "\n";

cout << firstNum << secondNum << "\n";

// Call the function. will change the values of firstNum and secondNum

swapNums(firstNum, secondNum);

cout << "After swap: " << "\n";

cout << firstNum << secondNum << "\n";

return 0;

}

**Part 3: Function Overloading**

A program that reads two integers and two characters. Overload the function larger to be able to compare the two integers and also compare the two characters. Use the following prototypes:

int larger(int x, int y);

char larger(char first, char second);

#include<iostream>

using namespace std;

int larger(int x, int y);

char larger(char first, char second);

int main()

{

int a,b;

char c,d;

cout<< "Enter two integers\n";

cin>>a>>b;

cout<<"The larger number is: "<<larger(a,b)<<endl<<endl;

cout<<"Enter two characters\n";

cin>>c>>d;

cout<<"The larger character is: "<<larger(c,d)<<endl<<endl;

}

int larger(int x, int y)

{

if(x > y)

return x;

return y;

}

char larger(char x, char y)

{

if(x > y)

return x;

return y;

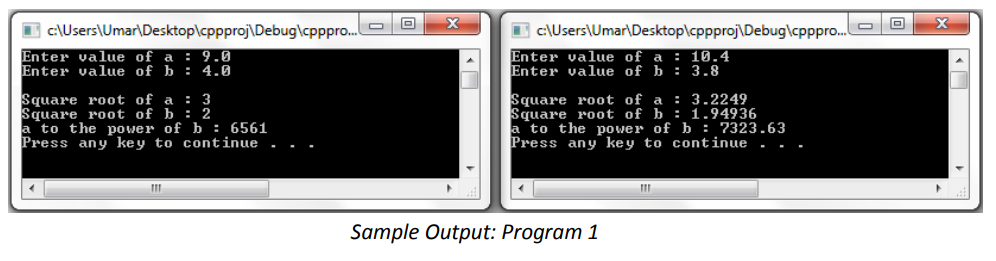
}

**EXERCISES**

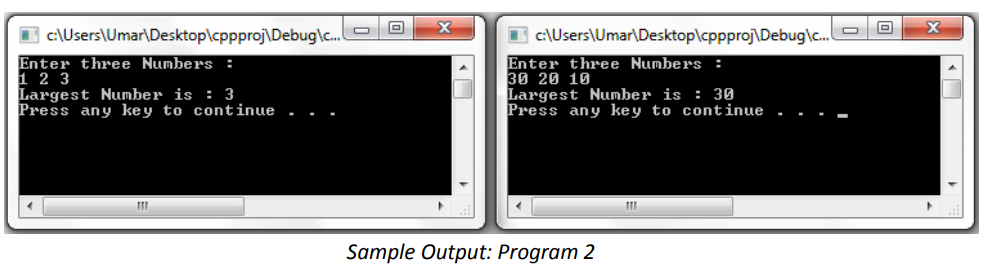
**Predefined Functions & User defined Functions**

**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.



**Program 2:** Write a function to 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.



**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

**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

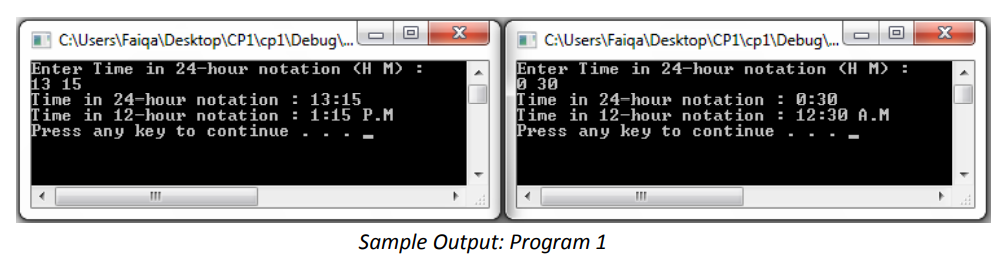
Description automatically generated

**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.

****

**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**

**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**