King Saud University
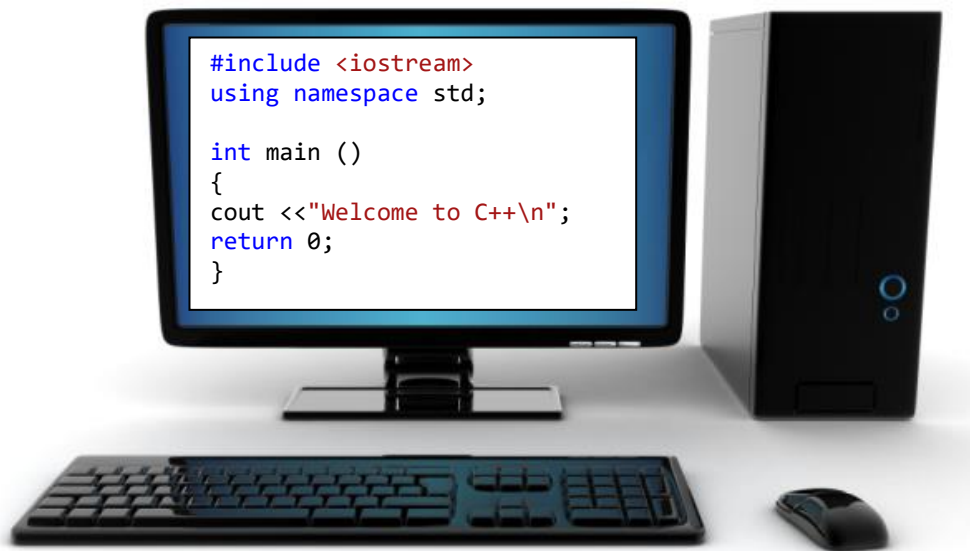
College of Engineering, Electrical Engineering Department

Labwork Manual:
GE 211 Computer Programming in "C++"

```cpp
#include <iostream>
using namespace std;

int main ()
{
cout <<"Welcome to C++\n";
return 0;
}
```

Student Name:

ID Number   :

2015

Pre-Lab: Getting Started

Objective   **:** Learn how to make a simple C++ program using Microsoft Visual Studio ٢٠10

Problem   **:** Make a simple C++ program to display a text "Welcome to C++ Programming" to the monitor.

Guideline to make new project in Visual Studio 2010:

1- Start the compiler by choosing **Start > Program Files > Microsoft Visual Studio ٢٠10**

2- Make a new project, go to **File > New > Project > Visual C++ > Win32 > Win32 console application.** In the file name area, write "*lab1*"**.** Click OK. A window (see Fig. 1) will appear, just click *next*, then select *console application* and *empty project*. Click *Finish*.

3- To write a source code for the project **lab1**, go to *Solution Explorer* windows, then click **lab1 > Source Files** (right click) **> add > new item**, and select *C ++ File (.cpp)*, then write "*sourcecode1*" in the file name area. Then click OK (see Fig. 2).
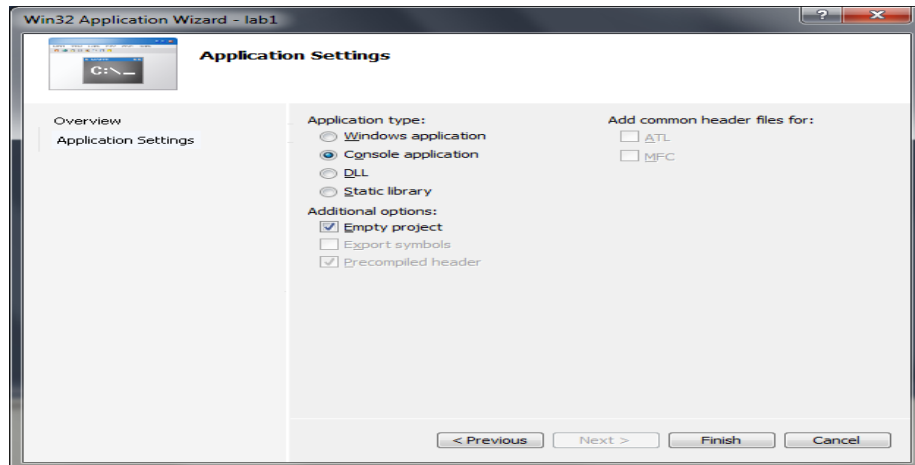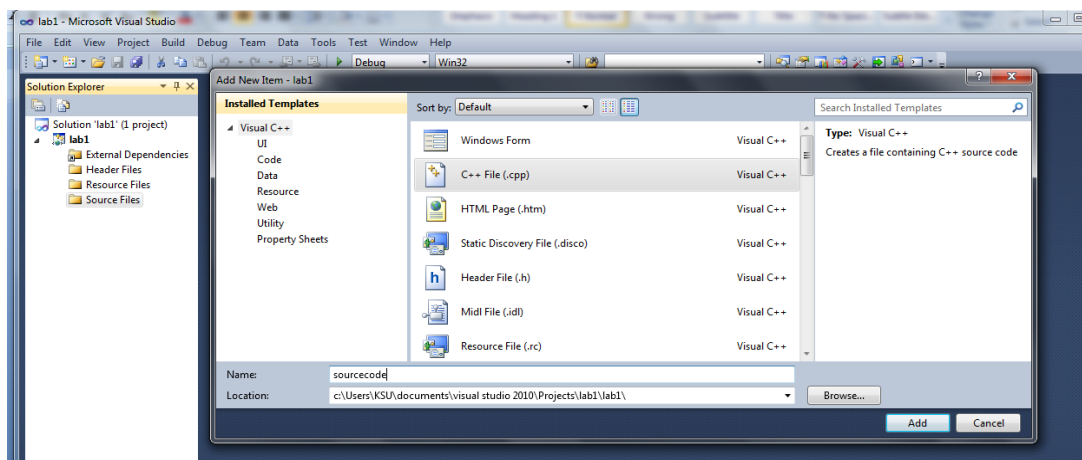
Fig. 1

Fig. 2

4- Type the following code (Fig. 3).

```
#include <iostream>
using namespace std;

int main ()
{
cout << "Welcome to C++ Programming\n";
return 0;
}
```

Fig. 3

5- After finishing writing the code, check if there is any error. You can compile the code to check the error (click *build*, then click *compile*). If there is no error after compiling then build your project by click *build*, then *build solution*.

If successful (no error) then run your program. To run the program click *debug*, then click *start without debugging*. Figure 4 should appear in your screen.
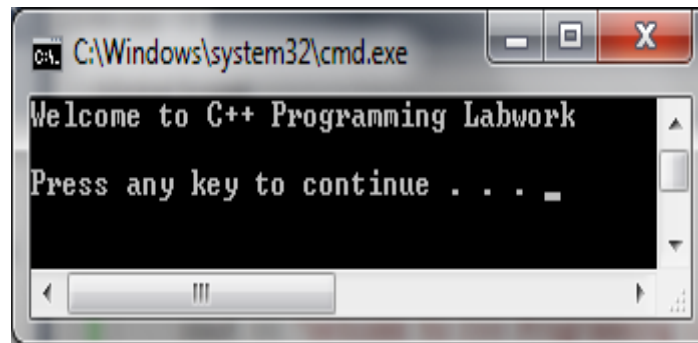

Fig. 4

Important note: Make a copy of this code in your flash as backup (also in the next Labworks, don't forget to make a copy of your file(s)).

Lab (1)

Objectives of this lab:

- Learn about software life cycle

- Illustration of the problem solving and programming - A simple C++ program

- Learn about variables and assignments

Exercise 1:

Type in the following program, be careful C++ is case sensitive:

```cpp
// C++ Lab 1 Exercise 1
//
// Program by: Place your name here

#include <iostream>
using namespace std;
int main()
{
cout << "Hello World!";
return 0;
}
```

Exercise 2:

write a program that converts degrees Celsius (TC) to degree Fahrenheit (TF) (Recall that TF = (9/5) TC + 32)
* Choose the suitable variables type.
* The program asks the user to enter the degree Celsius (TC).

```cpp
// C++ Lab 1 Exercise 2
// A program to convert from TC to TF.
//
// Program by: Place your name here

#include <iostream>
using namespace std;
int main()
{
Float Celsius;
Float Fahrenhiet;
Cout<<"Enter Celsius temo:";
Cin>>Celsius;
Fahrenhiet=(5/9)*(Celsius+32);
Cout<<"fahrenhiet:"<<fahrenhiet<<endl;
Return 0;
}
.................
return 0; }
```

Post-lab:

1. Write a C++ program to produce an output that looks like the following.

```
My name is YourFirstName YourLastName

            Hello
        Hello       Hello
    Hello       Hello       Hello
        Hello       Hello
            Hello

Bye
```

2. What is the answer to the following arithmetic operations?

       a) *(1/5) * 3*
       b) *(8/2) * 4*
       c) *(1.0/5.0) * 3*

check your answer by writing simple code.

Lab (2)

Objectives of this lab:

- More about math functions

- Learn about Simple *if* and *if-else* statement.

- Simple loop mechanism.

-

Exercise 1:

write a C++ program to evaluate the function written below.

$$y = \begin{cases} x^2 + \sqrt{x} & for(x \geq 0) \\ 1 - e^x & for(-10 < x < 0) \\ |x+5| & for(x \leq -10) \end{cases}$$

```
#include <iostream>
using namespace std;
int main()
{ double x,y;

cout << " Enter the value of x";
cin >> x;

if (x >= 0 )
y= x*x + sqrt(x);
else if ( x > -10 && x < 0 )
y= 1 - exp (x);
else
y= fabs ( x + 5 );

cout << " y = " <<y<<endl;

return 0;
}
```

Exercise 2:

Read and average 6 integers using while loop, print the result.

```
#include <iostream>
using namespace std;
int main()
{
    int x;
    int count = 0;
    double sum = 0;
    double average;

    // prompt the user:
    cout << "Enter six grades separated by a
single space, then press <Enter>: ";
    while( count < 6)
    {
        cin >> x;
        sum = sum + x;
        count++;
    }
    cout << endl;

    average = sum/6;
    cout << "The average is " << average <<
endl;

    return 0;
}
```

Post-lab :

1. Determine the value of each of the following arithmetic expressions. (when embedded in a small C++ program)

arithmetic expressions	Answer	arithmetic expressions	Answer
sqrt(16.0)		pow(2,3)	
sqrt(16)		abs(-3)	
pow(2.0,3.0)		fabs(-3.5)	
ceil(5.1)		floor(5.8)	
(7+sqrt(4.0))/3.0		pow(3.0,2)/2	

2. Modify exercise 2 that allow user to read and average **N** integers, print the result.	3. Change exercise 2 to work with a *do while loop* instead of *while loop*

Lab (3)

Objectives of this lab:

- Learn about Simple *if* and *if-else* statement.

- Simple loop mechanism.

- Basic conditional loop thinking

Exercise 1: Cola Machine

Write a program that presents the user w/ a choice of your 5 favorite drinks: (Coke, Water, Sprite, Juice, and Milk).

Then allow the user to choose a drink by entering a number 1-5.

Output which drink they chose.

- Modify the program so that if the user enters a choice other than 1-5 then it will output "Error. Choice was not valid."

```cpp
// C++ Lab 2 Exercise 1: Cola Machine
//
// Program by: Place your name here

#include <iostream>
using namespace std;
int main()
{
int choice;

cout<<"Please select a drink: 1)Coke 2)Water 3)Sprite 4)Juice
5)Milk"<<endl;
cin>>choice;

if ( choice == 1)  cout<< "Coke" <<endl;
if ( choice == 2)  cout<< "Water" <<endl;
if ( choice == 3)  cout<< "Sprite" <<endl;
if ( choice == 4)  cout<< "Juice" <<endl;
if ( choice == 5)  cout<< "Milk" <<endl;


return 0;
}
```

Exercise 2: while loop Exercise
Write a program that continues to ask the user to enter any number other than 5 until the user enters the number 5.
Then tell the user "Hey! You weren't supposed to enter 5!" and exit the program.

```cpp
// C++ Lab 2 Exercise 2  While loop
//
// Program by: Place your name here

#include<iostream>
usingnamespace std;
int main()
{
int number;


cout<<"Enter a number to start the game"<<endl;
cin>> number;
cout<<" You have entered the number  "<< number <<endl;

while (number != 5)
{

cout<<" Enter a number other than the number "<< number <<endl;
cin>> number;
cout<<" You have entered the number  "<< number <<endl;
}

cout<<endl<<" Hey! You weren't supposed to enter 5!"<<endl;
cout<<"  End of the game :: Good BYE \n";


return 0;
}
```

Post-lab :

Q1.Modify Exercise 1 to use a *switch statement* instead of *if statements*.

Q2. Modify Exercise 2 to calculate the summation (total) of all numbers entered by the user.

Q3. Modify Exercise 2 so that after 10 iterations if the user still hasn't entered 5, the program will tell the user "Finished." and exit.

Lab (4)

Objectives of this lab:

- Learn about for loop.

- A break statement in a for loop.

- Explicitly nested loops.

Exercise 1:

Write a program that finds the summation of the even numbers 0 to n (
where n is an even number greater than 0).
- Modify the program that will enable the user to find the summation from k
to n (where k < n both are even numbers).

```cpp
// C++ Lab 4 Exercise 1
// Program by: Place your name here

#include <iostream>
using namespace std;
int main()
{
int n;
cin>> n;
int sum = 0;

for (inti = 0; i<= n; i=i+2)  //Note that the variable i is a local
sum = ?????????;              //variable of the body of the for loop!

cout<< "The sum of the even numbers from 0 to "<<n<<" is "
<<sum<<endl;
return 0;
}
```
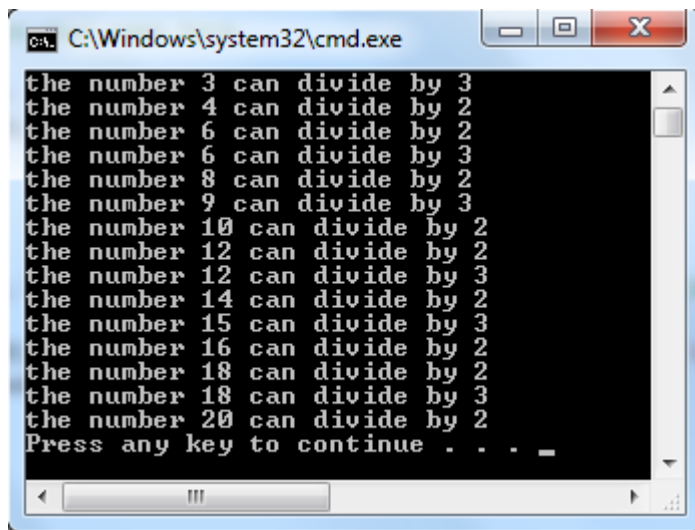
Exercise 2:

Write a program that prints out all the numbers between 3 and 20 that
divides by 2 and 3.

```cpp
// C++ Lab 4 Exercise 2
// Program by: Place your name here

#include<iostream>
usingnamespace std;
int main()
{

for (inti = 3; i<= 20; i++)  //Note that the variable i is a local
{
if ((i%2) ???????) cout<<" the number "<<i<<" can divide by 2 \n";
if ((i%3) ???????)    cout<<" the number "<<i<<" can divide by 3\n ";


}
return 0;
}
the output
```

Post-lab

Q1.

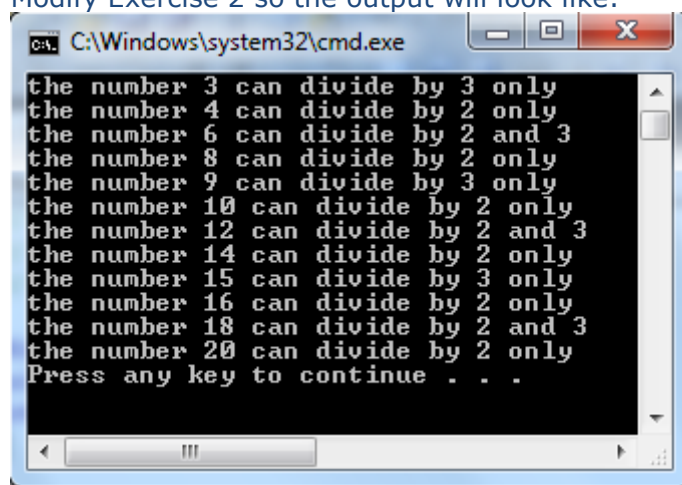Modify Exercise 1 so that it calculates the following:

sume : is the summation of all even numbers between k and n

sumo : is the summation of all odd numbers between k and n

sum   : is the summation of all numbers between k and n

Q2.
Modify Exercise 2 so the output will look like:

Lab (5)

Objectives of this lab:
- Learn about Functions that return a value.
- Learn about the importance of argument ordering.
- Learn about functions which return more than one value (call_by_reference).

Exercise 1:
This program computes the total cost of purchases made, including 5% sales tax, on number items at a cost of price each.

```cpp
#include <iostream>
using namespace std;

doubletotal_cost(int number, double price);     // (1) Function
declaration

int main( )
{
        double p, bill;
        int k;

        cout<< "Enter the number of items purchased: ";
        cin>> k;
        cout<< "Enter the price per item $";
        cin>> p;

        bill = total_cost(k, p);                    // (2) Function call
        cout.setf(ios::fixed);
        cout.setf(ios::showpoint);
        cout.precision(2);

        cout<< k << " items at "  << "$" << p<< " each.\n"
                << "Final bill, including tax, is $" << bill  <<endl;

        return 0;
}
// (3) Function definition
doubletotal_cost(int number, double price)          // Function
heading
{
        const double TAX_RATE = 0.05;
        double subtotal;
        subtotal = price * number;
      subtotal = subtotal + subtotal*TAX_RATE
        return (subtotal);
        }
```

Exercise 2:
Use function to compute the side area and the cross section area of a cylinder

```cpp
#include<iostream>
#include<cmath>
using namespace std;
const double PI = 3.14159;

double area(double r);  // Function that computes cross section area

double area(double r, double h);  // Function that computes side area
```

```
int main()
{
     double h, r, c_area, s_area;  //variables local to the main
function

     cout<< "Enter the radius and the height of a cylinder ";
     cin>> r >> h;
     cout<<endl;
c_area = area(r);
     cout<< "The cross section area of the cylinder is "<<c_area<<endl;
s_area = area(r, h);
     cout<< "The side area of the cylinder is "<<s_area<<"  \n\n";

     return 0;
}

doublearea(double r)
{
    double a;
a  = (2*PI*pow(r,2));
return a;
}
double area(double r, double h)
{
     double area;
     area = 2*PI*r*h;
     return area;
}
```

Exercise 3:

The function *Process (int a, int b, int c, int& s, double&g)* computes the sum and average of 3 integer numbers. Write a C++ program that calls the function *Process*

```
Solution:
#include <iostream>
using namespace std;
void  Process (int a, int b, int c, int& s, double& g);     // (1)
Function declaration
int main( )
{
     double  avg;
     int  x1,x2,x3,sum ;
cout<< " please enter 3 integer numbers \n";
cin>> x1 >> x2 >> x3 ;

     . . . . .           // (2) Function call
     . . . . .
     cout>> . . .
     return 0;
}
// (3) Function definition
void  Process (int a, int b, int c, int& s, double& g)// Function
heading
{
    s = a + b + c;
     g = s/3.0;

}
```

Post-Lab:

Q1. Write a separate function to calculate the tax called:

```
doubletaxed_price (double   price)
```

And then modify the code of Ex.1.

Q2. Modify the function of exercise 3 to be of the form:

double Process (int a, int b, int c, int& s)

Lab (6)

Objectives of this lab:

- Infinite series implementation.

- Random number implementation.

- Time function

Exercise 1: Estimation of π using Leibniz series

Write a program to calculate the value of (pi 'π')using Leibniz series.

$$\frac{\pi}{4} = \sum_{k=0}^{\infty} \frac{(-1)^k}{2k+1}.$$

```
// C++ Lab 6 Exercise 1  Estimation of pi using Leibniz series
//
// Program by: Place your name here

#include <iostream>
#include <cmath>
#define PI 3.14159265359

using namespace std;
int main()
{
     double p = 0.0;
     doublex,error;
     int N;

     cout<< "Enter the number of N? ";
     cin>> N;
     cout<<endl;
     p = 0.0 ;
for(int k= 1 ; k<=N ; ++k)
          {
          x = -pow(-1,k) / (2*k-1);
          p += x ;
     }
     p = 4*p;
     error = fabs(PI - p) ;
cout<< "N = "<< N <<"  pi is  "<< p <<" the error is "<< error <<"\n the
percent error is %"<< 100*(error/PI) <<endl;

     return 0;
}
```

```
Enter the number of N? 10

N = 10  pi is  3.23232 the error is 0.0907232
 the percent error is %2.88781
```

Exercise 2: Guessing Game

Write a program that implements a guessing game. The program will generate a random number between 1 and 100 inclusive.

```cpp
/* C++ Lab 6 Exercise 2  guessing game
   Program by: Place your name here
*/

#include<iostream>
#include<cstdlib>
#include<ctime>

using namespace std;

int main()
{
      srand(time(0));
      int number=rand()%100+1;
      int guess, tries(1);

      cout<<"Im thinking of a number between 1-100. Take a guess: ";
      cin>>guess;


      while(guess!=number)
      {
            tries++;

            if(guess>number)
                  cout<<"Too high,Guess again: ";
            else
                  cout<<"Too low, Guess again: ";

            cin>>guess;
      }
      cout<<"Congrats!! You got it after "<< tries << " tries." <<endl;


      return 0;
}
```

```
Im thinking of a number between 1-100. Take a guess: 50
Too low, Guess again: 80
Too low, Guess again: 90
Too high,Guess again: 85
Too low, Guess again: 88
Congrats!! You got it after 5 tries.
```

Post Lab

Q1. Modify Ex. 1 so that it finds N that gives any given error limit.

Q2. Use the time function from Ex2 to generate 5 beeps every 10 seconds for 3 times.

Lab (7)

Objectives of this lab:

- Learn about I/O streams.

- File name as Input.

- End of file (eof) and appending to file (app).

Exercise 1:

From Lab 1 we will recall this problem:

Write a program that converts degrees Celsius (TC) to degree Fahrenheit (TF) (Recall that TF = (9/5) TC + 32)

* Choose the suitable variables type.

* The program reads TC from input file "input_temp.dat" and writes TF to the file "out_temp.dat"

(Open a notepad file and store the Celsius temperature then save your input file with "input_temp.dat)

Part of the code is given below:

```
// C++ Lab 8 Exercise 1

#include <iostream>
#include <f………….>

using namespace std;
int main()
{
double TC,TF;
ifstream xxx;
xxx.open (" input_temp.dat");
ofstreamyyy;
yyy.open ("out_temp.dat");

xxx>> TC;

TF = . . . . .

yyy<<TC << "    " <<TF;


return 0;
}
```

Exercise 2:

If the file circuit.dat contains the following data

3.0	2.1
1.5	1.1
2.6	4.1

The first column is voltage and the second column is the electric current.

Write program that reads the voltages and currents then calculates the electric power (P) based on the equation:

$$P = v * i$$

Write your output to the file results.dat with voltage in the first, current in the second and power on the third column.

Your output file should look like

Voltage	Current	Power
3.0	2.1	(result)
1.5	1.1	(result)
2.6	4.1	(result)

Solution:

```
#include <iostream>
#include <fstream>
using namespace std;
int main ()
{
ifstreamaaa;
aaa.open ("circuit.dat");
ofstreamppp;
ppp.open("result.dat");
inti;
doublea,b,p;
ppp<< "Voltage"<<"  Current"<<"  Power"<<endl;
for (i=1; i<=3; i++)
{
aaa>>a>>b;
p=b*a;
ppp<< a <<"   "<<b <<"   "<< p<<endl;

}
return 0;
}
```

Post-Lab:

Q1.   Run your code in EX.1 again but use the file name as input using the following statements:

```
char file[15];
cin>> file;
xxx.open(file)
```

Q2.   If the circuit.dat file in EX.2 has header line that is look, like

v	i
3.0	2.1
1.5	1.1
2.6	4.1

Modify your program to read the new circuit.dat file and run your program twice with keeping the previous results (use append function)

Q3.   If you do not know the number of lines in the circuit.dat filein EX.2, modify your program to read the input file by using eof function.

Lab (8)

Objectives of this lab:
- Learn about Arrays.
- Using Arrays in function calls.
- Programming with arrays.

Exercise 1:
Repeat Exercise 2 from lab7 using arrays for voltage, current and power instead of single value variables.

Solution:
```
#include <iostream>
#include <fstream>
using namespace std;
int main ()
{
ifstreamaaa;
aaa.open ("circuit.dat");
ofstreamppp;
ppp.open("result.dat");
inti;
double a[3],b[3],p[3];
ppp<<" Voltage"<<" Current"<<" Power"<<endl;
for (i=1; i<=3; i++)
{
aaa>>a[i]>>b[i];
p[i]=b[i]*a[i];
ppp<< a[i]<<"   "<<b[i]<<"   "<< p[i]<<endl;

}
return 0;
}
```

Post lab Q1:
Calculate power in Exercise 1 by calling function named power which has the below definition:
void power (double v[ ] , int lv, double i[ ] , int li)
Where
> lv is the length of voltage array (# of voltages)
> li is the length of current array (# of currents)

Exercise 2:
Write a program that able to sort a list of values stored in a given array.
- Complete the given functions in the code so that the *sort* function is sorting the values in an ascending order (ترتيب تصاعدي من الأصغر للأكبر.).

```
//Sorting an Array
#include<iostream>

void sort(int a[], intarray_size);
//Postcondition: The values of a[0] through a[number_used - 1] have
//been rearranged so that a[0] <= a[1] <= ... <= a[number_used - 1].

voidswap_values(int& v1, int& v2);
//Interchanges the values of v1 and v2.
```

```cpp
intindex_of_smallest(constint a[], intstart_index, intarray_size);
//Returns the index i such that a[i] is the smallest of the values

int main( )
{
usingnamespacestd;
constint size =10;
intarr[size] = {68,70,21,4,90,34,21,61,13,2};

sort(arr, size);
cout<<"In sorted order the numbers from lowest to highest (ascending order):\n";
for (int index = 0; index < size; index++)
cout<<arr[index] <<" ";
cout<<endl;
return 0;
}

void sort(inta[], intarray_size)
{
ints_index;
for (inti = 0; i<array_size - 1; i++)
        {
s_index=  index_of_smallest(a, i, array_size);

swap_values(a[i], a[s_index]);
    }
}

voidswap_values(int&v1, int&v2)
{
int temp;
temp = v1;
    v1=v2;
    v2=temp;
      // complete here...
}

intindex_of_smallest(constinta[], intstart_index, intarray_size)
{
      intorde, min= a[start_index];
order = start_index;
for (int k=start_index+1;  k<number_used; k++)
if(a[k]<min)
          {
// complete here...
min = a[k];
order = k;
          }

return order;
}
```
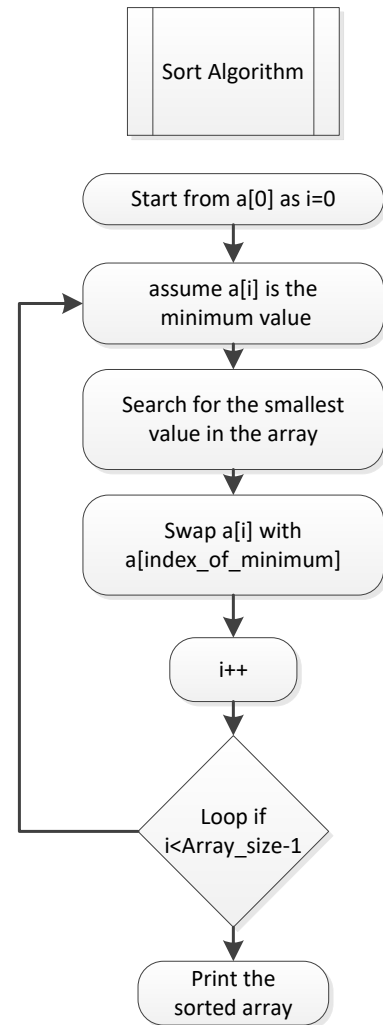
Flowchart (right side):

Sort Algorithm

Start from a[0] as i=0

assume a[i] is the minimum value

Search for the smallest value in the array

Swap a[i] with a[index_of_minimum]

i++

Loop if i<Array_size-1

Print the sorted array

Post lab Q2:

Write a new function called *sort_des ()* to sort the values in a descending order (ترتيب تنازلي من الأكبر للأصغر.)

Lab (9)

Objectives of this lab:

- More about Arrays.

- Multidimensional Arrays.

Exercise 1:

For the given matrix

$$h = \begin{bmatrix} 2 & -4 & -1 & 3 & 2 \\ 3 & 2 & 0 & -2 & 6 \\ 4 & 6 & 0 & -1 & 6 \end{bmatrix} \begin{matrix} \text{row 1} \\ \text{row 2} \\ \text{row 3} \end{matrix}$$

Write a C++ program with the following requirements (all matrix elements are integers)
1. Enter all matrix elements column by column from the keyboard.
2. Find the summation for second row (sum2).
3. Find the maximum number in the matrix (max).
4. Find the maximum number in the matrix (min).

Post lab Q1:

1. Find the transpose of the matrix (ht). (replace rows with columns and columns with rows)
2. Find the number of elements that are greater than 0.

Exercise 2:

Write a function named "subtotal" takes as its arguments the following:
(1) an array of floating point values;
(2) an integer that tells the number of cells in the array.
The function should replace the contents of each cell with the sum of the contents of all the cells in the original array from the left end to the cell in question.

Use the following array to test your program

x[8] =

2.3	3.7	6.0	-4.0	0	-7.5	1	1.5

After calling the function, the output should look like:

x[8] =

2.3	6.0	12.0	8.0	8.0	0.5	1.5	3.0

Solution:

```
# include <iostream>
using namespace std;
void subtotal (float a[], int n);

int main()
```

```
{        float   x[8];
         for (inti=0 ; i<8; i++)
         {
                  cin>> x[i];
         }
```

Print out array x to the screen before calling the function

Function call here

Print out array x to the screen after calling the function

```
return 0;
}

void subtotal (float a[], int n)
{
inti;

for (i = 1; i< n; ++i)
a[i] += a[i-1];
}
```

Post lab Q1:

Write a function named "rotate_right" that takes as its arguments the following:
(1) an array of floating point values;
(2) an integer that tells the number of cells in the array;
The function should shift the contents of each cell one place to the right, except for the contents of the last cell, which should be moved into the cell with subscript 0

For example:
x[8] =

2.3	3.7	6.0	-4.0	0	-7.5	1	1.5

After calling the function, the output should look like:

x[8] =

1.5	2.3	3.7	6.0	-4.0	0	-7.5	1