

Lab Task #1

NAME: Raheel Nazir

SAP: 54513

Q1- Write a program in C++ that take integer type 2D array from user, calculate sum multiplication and average of all numbers.

```
int row;
cout <<"Enter No of Rows: ";
cin>>row;
int col;
cout <<"Enter No of Columns: ";
cin>>col;

int twoDArray[row][col];

for(int i=0; i<row; i++) {
    for(int j=0; j <col; j++) {

        cin>>twoDArray[i][j];

    }
    cout <<endl;
}

int sum, multiplication = 1, average ;
```

```

for(int i=0; i<row; i++) {
    for(int j=0; j <col; j++) {

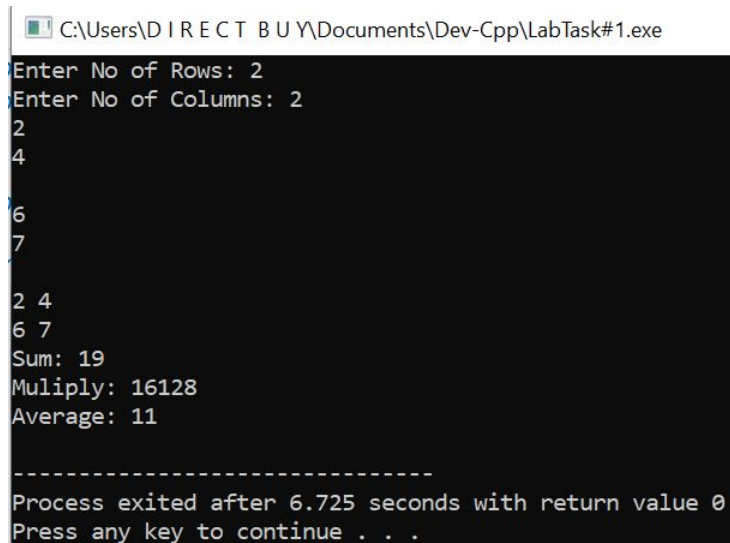
        cout<<twoDArray[i][j] << " ";

        sum = sum + twoDArray[i][j];
        multiplication = multiplication * twoDArray[i][j];
        average = sum / row + col;
    }
    cout <<endl;
}

cout << "Sum: " << sum << endl;
cout << "Muliply: " << multiplication <<endl;
cout << "Average: " << average << endl;

```

Output:



C:\Users\DIRECTBUY\Documents\Dev-Cpp\LabTask#1.exe

```

Enter No of Rows: 2
Enter No of Columns: 2
2
4
6
7
2 4
6 7
Sum: 19
Muliply: 16128
Average: 11

-----
Process exited after 6.725 seconds with return value 0
Press any key to continue . . .

```

Q2- Write a program in C++ to swap values of two variables using pointers.

```
int a = 2;
int b = 1;

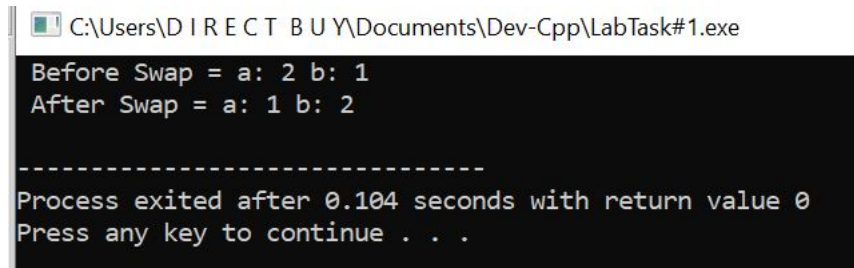
cout << " Before Swap = " << "a: " << a << " " << "b: " << b << endl;

int *ptr = &a;
int *btr = &b;

int c = a;
a=b;
b = c;

cout << " After Swap = " << "a: " << a << " " << "b: " << b << endl;
```

Output:



```
C:\Users\DIRECTBU\Documents\Dev-Cpp\LabTask#1.exe
Before Swap = a: 2 b: 1
After Swap = a: 1 b: 2

-----
Process exited after 0.104 seconds with return value 0
Press any key to continue . . .
```

Q3- Write a program that lets the user to enter the 10 values into the array. The program should then display the largest and the smallest values stored in the array.

```

int n =10;

cout <<"Enter ten Values: ";

int array[n];

for(int i=0; i<n; i++) {
    cin>>array[i];
}

int max = array[0];

for(int i=0; i<n; i++) {
    if(array[i]>max) {
        max = array[i];
    }
}

cout << "Largest Number in Array: " << max <<endl;

int min = array[0];

for(int i=0; i<n; i++) {
    if(array[i]<min) {
        min = array[i];
    }
}

cout << "Smallest Number in Array: " << min <<endl;

```

Output:

C:\Users\DIRECT B U Y\Documents\Dev

```

Enter ten Values: 12
33
54
65
76
8
67
87
65
54
Largest Number in Array: 87
Smallest Number in Array: 8

```

Q4- Write a program that lets the user to enter the total rainfall for each of 12 months into an array of doubles. The program should calculate and display the total rainfall for the year, the average monthly rainfall and the month with the highest and lowest rainfall.

```
int n = 12;
int counter = 1;
cout << "Enter Each Month Rainfall: " << endl;

double totalRainfall = 0;
double averageRainfall;
double monthlyRainfall[n];

for(int i=0; i<n; i++) {
    cout << "Month " << counter << " = " ;
    cin >> monthlyRainfall[i];
    counter ++;
    totalRainfall = totalRainfall + monthlyRainfall[i]; // total rainfall
    averageRainfall = totalRainfall / n ;           // average rainfall
}

cout << "Total Rainfall for the Year " << " = " << totalRainfall << "mm" << endl;
cout << "Monthly Average Rainfall " << " = " << averageRainfall << "mm" << endl;
```

Output:

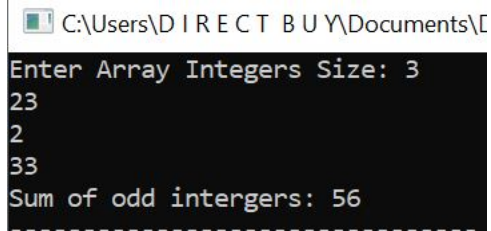
```
C:\Users\DIRECTBUY\Documents\Dev-Cpp\LabTask
Enter Each Month Rainfall:
Month 1 = 23.4
Month 2 = 54.3
Month 3 = 54
Month 4 = 65
Month 5 = 100
Month 6 = 12.9
Month 7 = 234.5
Month 8 = 432.5
Month 9 = 4.3
Month 10 = 233
Month 11 = 22
Month 12 = 100
Total Rainfall for the Year = 1335.9mm
Monthly Average Rainfall = 111.325mm
```

Q6- Write a program that dynamically allocates an array of integers. Read the values from user and calculate the sum of odd integers.

```
int n;
cout << "Enter Array Integers Size: ";
cin>>n;
int array[n];
int oddSum;
for(int i=0; i<n; i++) {
    cin>>array[i];
    if(array[i]%2 != 0) {
        oddSum = oddSum + array[i];
    }
}
```

```
cout << "Sum of odd intergers: " << oddSum ;
```

Output:

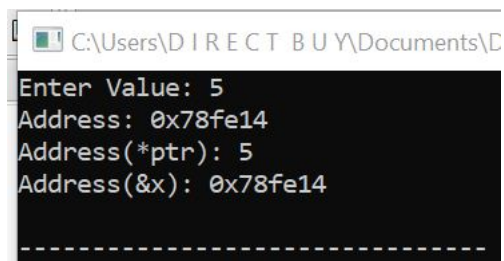


A screenshot of a Windows command prompt window. The title bar shows the path 'C:\Users\DIRECTBUY\Documents\'. The prompt displays the following text: 'Enter Array Integers Size: 3', followed by three lines of input: '23', '2', and '33'. The final output line is 'Sum of odd intergers: 56'. The window ends with a dashed line separator.

Q7- Define a pointer variable. Assign the address of variable to a pointer variable and access the value of address variable in the pointer variable.

```
int *ptr;  
  
int x;  
cout << "Enter Value: ";  
cin >> x;  
  
ptr = &x;  
  
cout << "Address: " << ptr << endl;  
cout << "Address(*ptr): " << *ptr << endl;  
cout << "Address(&x): " << &x << endl;
```

Output:

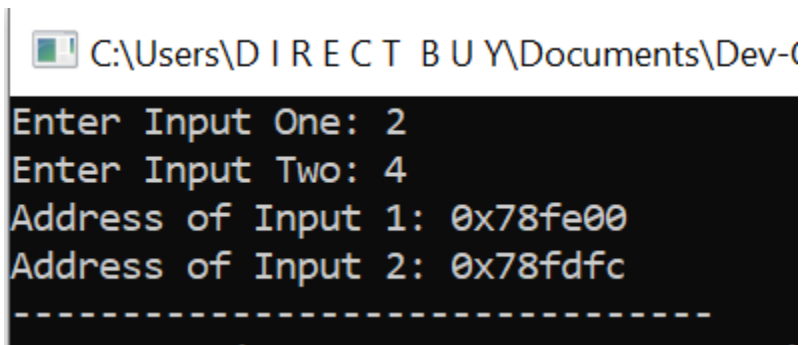


A screenshot of a Windows command prompt window. The title bar shows the path 'C:\Users\DIRECTBUY\Documents\'. The prompt displays the following text: 'Enter Value: 5', followed by four lines of output: 'Address: 0x78fe14', 'Address(*ptr): 5', and 'Address(&x): 0x78fe14'. The window ends with a dashed line separator.

Q8- Write a program that asks the user to enter integers as inputs to be stored in the variables 'a' and 'b' respectively. There are also two integer pointers named ptrA and ptrB. Assign the values of 'a' and 'b' to ptrA and ptrB respectively, and display them.

```
int input1;  
int input2;  
  
cout << "Enter Input One: " ;  
cin >> input1;  
cout << "Enter Input Two: " ;  
cin >> input2;  
  
int a = input1;  
int b = input2;  
  
int *var1 = &a;  
int *var2 = &b;  
  
cout << "Address of Input 1: " << var1 << endl ;  
cout << "Address of Input 2: " << var2;
```

Output:



```
C:\Users\DIRECTBU\Documents\Dev-  
Enter Input One: 2  
Enter Input Two: 4  
Address of Input 1: 0x78fe00  
Address of Input 2: 0x78fdfc  
-----
```

**Q9- Write a program for a calculator using functions.
Your program must have the following functions:
a. Menu () telling the user to select from the options**

b. Addition (int a, int b) adding two numbers

c. Subtraction (int a, int b)

d. Division (int a, int b)

e. Multiplication (int a, int b)

f. Pow (int number, int pow)

In the main function you will only call the menu () function.

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

```
int a ,b;
```

```
void add(int a, int b) {
```

```
    cout <<"Enter Input 1: ";
```

```
    cin>> a;
```

```
    cout <<"Enter Input 2: ";
```

```
    cin>> b;
```

```
    int sum = a + b;
```

```
    cout << "Sum: " <<sum <<endl;
```

```
}
```

```
void sub(int a, int b) {
```

```
    cout <<"Enter Input 1: ";
```

```
    cin>> a;
```

```
    cout <<"Enter Input 2: ";
```

```
    cin>> b;
```

```
    int subt = a - b;
```

```
    cout << "Subtraction: " <<subt <<endl;
```

```
}
```

```
void mul(int a, int b) {
```

```
    cout <<"Enter Input 1: ";
```

```
    cin>> a;
```

```
    cout <<"Enter Input 2: ";
```

```
    cin>> b;
```

```
    int multiply = a * b;
    cout << "Multiplication: " << multiply << endl;
}
```

```
void div(int a, int b) {
    cout << "Enter Input 1: ";
    cin >> a;
    cout << "Enter Input 2: ";
    cin >> b;


    int division = a / b;
    cout << "Divison: " << division << endl;
}
```

```
void menu() {
    int choice;
    cout << "1. Addition" << endl;
    cout << "2. Subtraction" << endl;
    cout << "3. Multiplication" << endl;
    cout << "4. Division" << endl;

    while(true) {
        cin >> choice;
        switch(choice) {
            case 1:
                add(a,b);
                break;
            case 2:
                sub(a,b);
                break;
            case 3:
                mul(a,b);
                break;
            case 4:
                div(a,b);
                break;
        }
    }
}
```

```
    }  
}  
  
int main() {  
    menu();  
    return 0;  
}
```

Output:

 C:\Users\DIRECT BUY\Docu

```
1. Addition  
2. Subtraction  
3. Multiplication  
4. Division  
1  
Enter Input 1: 34  
Enter Input 2: 54  
Sum: 88  
3  
Enter Input 1: 34  
Enter Input 2: 5  
Multiplication: 170  
2  
Enter Input 1: 43  
Enter Input 2: 56  
Subtraction: -13
```

Q- 5. Write a program that creates a two dimensional array initialized with test data.

Perform the following operations on the array data.

Get total: get total of all the elements in the array.

Get Average: calculate the average of all the values in the array.

Get row total: calculate the total of all the values in the specified row.

Get column total: calculate the total of all the values in the specified column.

Get Highest in row: find the highest value in the specified row.

Get Highest in Column: find the highest value in the specified column.

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

```
int array[3][3] = {1 , 2, 3 ,
                   4, 5, 6 ,
                   7 , 8 , 9};
```

```
int row, col;
```

```
int rowTotal(int array[3][3], int row) {
```

```
    int total = 0;
    for (int i = 0; i < 3; i++) {
        total = total + array[row][i];
    }
    return total;
}
```

```
int columnTotal(int array[3][3], int col) {
```

```
    int total = 0;
    for (int i = 0; i < 3; i++) {
        total = total + array[i][col];
    }
    return total;
}
```

```

int highestRowElement(int array[3][3], int row) {
    int highest = array[row][0];

    for (int i = 1; i < 3; i++) {
        if (array[row][i] > highest) {
            highest = array[row][i];
        }
    }
    return highest;
}

```

```

int highestColumnElement(int array[3][3], int col) {

    int highest = array[0][col];

    for (int i = 1; i < 3; i++) {
        if (array[i][col] > highest) {
            highest = array[i][col];
        }
    }
    return highest;
}

```

```

int main() {

    int sum = 0;
    double average = 0;

    for (int i = 0; i<3; i++) {
        for(int j=0; j<3; j++) {

            sum = sum + array[i][j];

        }
    }
}

```

```

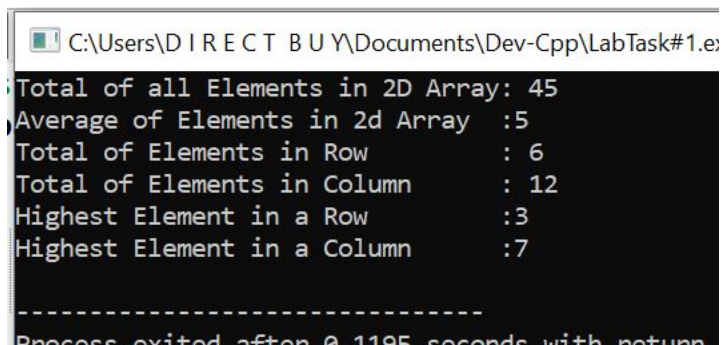
        average = (sum)/ 9;

        cout << "Total of all Elements in 2D Array: " << sum <<endl;
        cout << "Average of Elements in 2d Array :" <<average <<endl;
        cout << "Total of Elements in Row      : "<<rowTotal(array,row)
<<endl;
        cout << "Total of Elements in Column      :
"<<columnTotal(array,col) <<endl;
        cout << "Highest Element in a Row      : "
<<highestRowElement(array,row) << endl;
        cout << "Highest Element in a Column      : "
<<highestColumnElement(array,row) <<endl;

        return 0;
}

```

Output:



The screenshot shows a terminal window with the following output:

```

C:\Users\DIRECT B U Y\Documents\Dev-Cpp\LabTask#1.e
Total of all Elements in 2D Array: 45
Average of Elements in 2d Array :5
Total of Elements in Row      : 6
Total of Elements in Column      : 12
Highest Element in a Row      :3
Highest Element in a Column      :7
-----
Process exited after 0.1195 seconds with return

```