**Lab # 4**

**Week = 28th-April – 02nd-May-2025**

**Topics Covered:**

* Arrays
* Functions

**Problem Statement:**

Write a program in which you have to declare an integer array of size 10 and initializes it with numbers of your choice. Find the maximum and minimum number from the array and output the numbers on the screen.

For finding the maximum and minimum numbers from the array you need to declare two functions **findMax** and **findMin** which accept an array and size of array (an int variable) as arguments and find the max min numbers, and return those values.

**Solution:**

#include <iostream>

using namespace std;

//functions declaration

int findMin(int [],int);

int findMax(int [],int);

int main() {

const int SIZE = 10; //size of array

//Array initialization

int number[10] = {

21,25,89,83,67,81,52,100,147,10

};

//Displaying minimum and maximum number

cout<< "Maximum number in the array is :" <<findMax(number, SIZE) <<endl;

cout<< "Minimum number in the array is :" <<findMin(number, SIZE) <<endl;

return 0;

}

//Definition of findMin function

int findMin(int array[],int size){

int min = 0;

min = array[0];//Storing the value of the first element of array in 'min' variable

for (int i = 0; i<size; i++){ //loop for traversing array

if(min > array[i])//Testing if the value of 'min' variable is greater than the current element of array

min = array[i];//Storing the value of current element of array in 'min' variable

}

return min; //returning the minimum value of the array

}

//Definition of findMax function

int findMax(int array[],int size){

int max = 0;

max = array[0];//Storing the value of the first element of array in 'max' variable

for (int i = 0; i<size; i++){//loop for traversing array

if(max < array[i]) //Testing if the value of 'max' variable is less than the current element of array

max = array[i];//Storing the value of current element of array in 'max' variable

}

return max; //returning the maximum value of the array

}

**Lab # 3**

**Week = 21st-April – 25th-April-2025**

**Topics Covered:**

* Functions
* Repetition Structure (Loop)

**Problem Statement:**

Write a program in which you have to define a function displayDiagnol which will have two integer arguments named rows and cols. In the main function, take the values of rows and columns from the users. If the number of rows is same as numbers of columns then call the displayDiagnol function else show a message on screen that number of rows and columns is not same.

The following logic will be implemented inside the displayDiagnol function:

The function will take the value of rows and cols which are passed as argument and print the output in matrix form. To print the values in the matrix form, nested loops should be used. For each loop, you have to use a counter variable as counter. When the value of counters for each loop equals, then it prints the value of row at that location and prints hard coded zero at any other location.

Example if the user enters rows and cols as 3, then the output should be like this

1 0 0

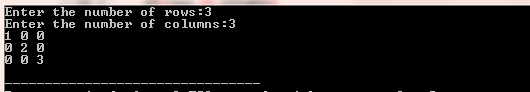
0 2 0

0 0 3

Example: when rows and columns are not same.



Example: when rows and columns are same.



**Solution:**

#include <iostream>

using namespace std;

void displayDiagonal(int,int); // function declaration

int main(){

int rows, columns;//variable declaration and initialization

rows = 0;

columns = 0;

cout<< "Enter the number of rows:";//Taking number of rows as input

cin>> rows;

cout<< "Enter the number of columns:";//Taking columns of rows as input

cin>> columns;

if(rows == columns)//conditional check for square matrix

displayDiagonal(rows,columns); // call function

else

cout<< "Wrong input! Num of rows should be equal to num of columns";

return 0;

}

// function definition

void displayDiagonal(int rows, int columns){

for (int i = 1; i<=rows; i++) {

for (int j = 1; j<=columns; j++){

if(i==j)//diagonal check

cout<<i<< " "; //displaying element

else

cout<< 0 << " ";

}

cout<< "\n";

}

}

**Lab # 2**

**Week = 14th-April – 18th-April-2025**

**Topics Covered:**

* Repetition Structure (Loop)

**Problem Statement: While Loop**

“Calculate the average age of a class of ten students using while loop. Prompt the user to enter the age of each student.”

* We need 10 values of variable age of int type to calculate the average age.

int age;

* “Prompt the user to enter the age of each student” this requires cin>> statement.

For example:

cin>> age;

* Average can be calculated by doing addition of 10 values and dividing sum with 10.

TotalAge = TotalAge + age1;

AverageAge = TotalAge / 10;

**Solution:**

#include<iostream>

using namespace std;

main() {

**// declaring variable age to take input**   
 int age=0;  
   
 **// declaring variables to calculate totalAge and averageAge**  
 int TotalAge = 0, AverageAge = 0;

**// declaring ageCounter variable to check the number of iterations**   
int ageCounter=0;  
  
**// comparing the value of ageCounter**

while (ageCounter <10)

{

cout<<"please enter the age of student "<<++ageCounter<<" :\t";

cin>>age;

**//calculate totalAge by adding age of all students**

TotalAge = TotalAge + age;

}

cout<< "Total Age of 10 students = "<<TotalAge<<endl;  
 **// calculate AverageAge by dividing the totalAge with number of students**  
 AverageAge = TotalAge/10;

**//Display the result (average age)** cout<<"Average of students is: "<<AverageAge;

}

**Lab # 1**

**Week = 14th-April – 18th-April-2025**

**Topics Covered:**

* Variables
* Data Types
* Arithmetic Operators
* Precedence of Operators

**Problem Statement:**

“Calculate the average age of a class of ten students. Prompt the user to enter the age of each student.”

* We need 10 variables of int type to calculate the average age.

int age1, age2, age3, age4, age5, age6, age7, age8, age9, age10;

* “Prompt the user to enter the age of each student” this requires cin>> statement.

For example:

cin>> age1;

* Average can be calculated by doing addition of 10 variables and dividing sum with 10.

TotalAge = age1 + age2 + age3 + age4 + age5 + age6 + age7 + age8 +age9 + age10 ;

AverageAge = TotalAge / 10;

**Solution:**

#include<iostream>

using namespace std;

int main(){  
  
**// declaring 10 integer variables to take input of students age**  
int age1,age2,age3, age4, age5, age6, age7, age8, age9, age10;  
  
**// declaring variables to calculate totalAge and averageAge**  
int TotalAge = 0, AverageAge = 0;

**// taking input of each student age**  
cout<<"please enter the age of student 1 ";

cin>>age1;

cout<<"please enter the age of student 2 ";

cin>>age2;

cout<<"please enter the age of student 3 ";

cin>>age3;

cout<<"please enter the age of student 4 ";

cin>>age4;

cout<<"please enter the age of student 5 ";

cin>>age5;

cout<<"please enter the age of student 6 ";

cin>>age6;

cout<<"please enter the age of student 7 ";

cin>>age7;

cout<<"please enter the age of student 8 ";

cin>>age8;

cout<<"please enter the age of student 9 ";

cin>>age9;

cout<<"please enter the age of student 10 ";

cin>>age10;

**//calculate totalAge by adding age of all students**  
TotalAge = age1 + age2 + age3 + age4 + age5 + age6 + age7 + age8 + age9 + age10;   
 **// calculate AverageAge by dividing the totalAge with number of students**AverageAge = TotalAge/10;

**//Display the result (average age)**

cout<<"Average of students is: "<<AverageAge;

}

**Alternative Solution**

#include<iostream>

using namespace std;

main() {  
  **// declaring variable age to take input** int age=0;  
   
 **// declaring variables to calculate totalAge and averageAge**  
 int TotalAge = 0, AverageAge = 0;

**// using for loop to take input of each students and adding them**  
 for (int i = 1;i<=10;i++){

cout<<"please enter the age of student "<<i<<" :\t";

cin>>age;

TotalAge += age;

}

**// calculate AverageAge by dividing the totalAge with number of students**  
 AverageAge = TotalAge/10;

**//Display the result (average age)**

cout<<"Average of students is: "<<AverageAge;

}