CS18-A

Lab 9

18F-0336

**Q1:**

Write a Program to read five numbers, find their sum, and print the numbers in reverse

order.

**Code:**

#include<iostream>

using namespace std;

int main() {

int arr[5];

int sum = 0, count;

for (count = 0; count < 5; count++)

{

cout << "Enter "<<count+1<<" Number= ";

cin >> arr[count];

sum = sum + arr[count];

}

cout << endl;

cout << "sum of numbers are :" << sum << endl;

cout << "numbers in reverse order are:";

for (count = 4; count >= 0; count--)

{

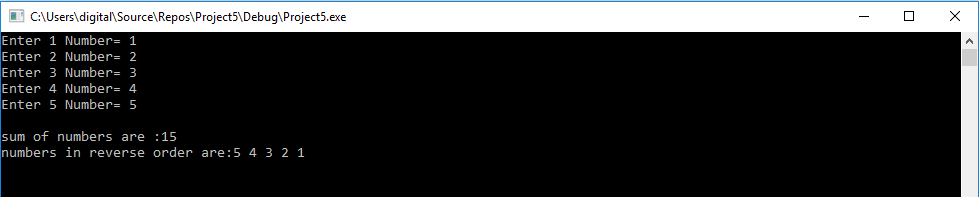
cout << arr[count] << " ";

}

system("pause>0");

return 0;

}



**Q2:**

**Write a Program to copy the elements of one array into another array of same size.**

**Code:**

#include <iostream>

using namespace std;

int main()

{

int arr1[10], arr2[10];

const int size = 10;

cout << " enter list to be copied : " << endl;

for (int i = 0; i < size; i++)

{

cout << "Enter Value: ";

cin >> arr1[i];

}

for (int i = 0; i < size; i++)

{

arr2[i] = arr1[i];

}

cout << "copied array = ";

for (int i = 0; i < size; i++)

{

cout << arr2[i] << " ";

}

system("pause>0");

return 0;

}



**Q3:**

**Write a Program to count the number of duplicate elements in an array.**

**e.g. given an array.**

**Array1[10]={2,5,2,6,9,9,8,4,2,1}**

**Solution:**

**Duplicates found:**

**2-&gt; 3 times**

**9-&gt; 2 times**

**Code:**

#include <iostream>

using namespace std;

int main()

{

int arr[10],repeat[10];

for (int i = 0; i < 10; i++)

{

cout << "Enter Values= ";

cin >> arr[i];

}

for (int i = 0; i < 10; i++)

{

int count = 1;

for (int j = i + 1; j < 10; j++)

{

if (arr[i] == arr[j])

{

count++;

}

}

if (count!=1)

{

cout << "Duplicate found! " << endl;

cout << "The number " << arr[i] << " is repeated " << count << " times" << endl;

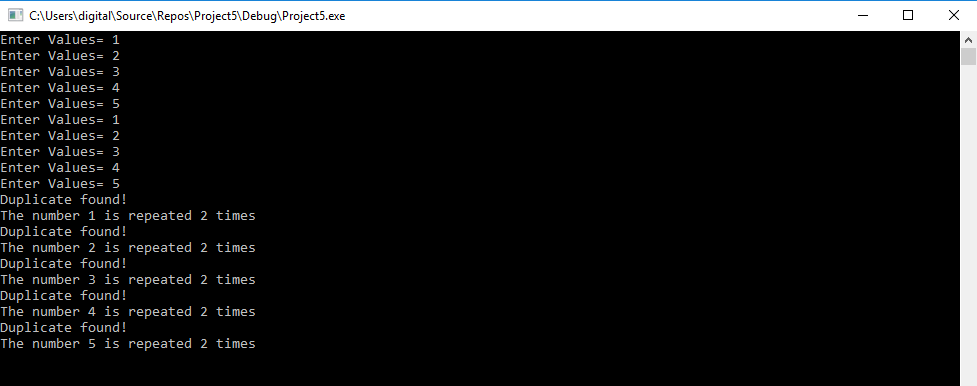
}

}

system("pause>0");

return 0;

}



**Q4:**

**Write a C++ program that declares an array alpha of 50 components of type double.**

**Initialize the array so that the first 25 components are equal to the square of the index**

**variable, and the last 25 components are equal to three times the index variable. Output**

**the array so that 10 elements per line are printed.**

**Code:**

#include <iostream>

using namespace std;

int main()

{

double arr[50];

for (int i = 0; i < 50; i++)

{

arr[i] = i \* i;

if (i > 25)

{

arr[i] = 3 \* i;

}

}

for (int i = 0; i < 50; i++)

{

if (i % 10 == 0)

{

cout << endl;

}

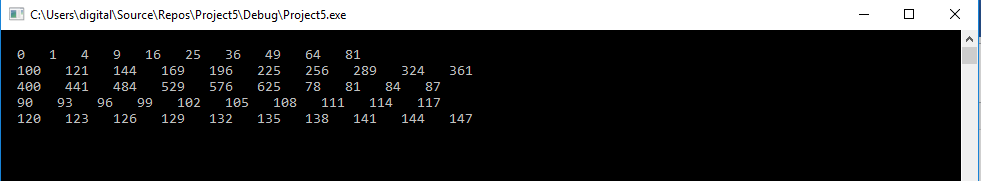
cout << " " << arr[i] << " ";

}

system("pause>0");

return 0;

}



**Q5:**

**Write a program in c++ to find the maximum and minimum element of a 1D array.**

**Code:**

#include<iostream>

using namespace std;

int main()

{

int small=0, large=0;

int arr[5];

cout << "Enter Values: " << endl;

for (int i = 0;i < 5;i++)

{

cin>> arr[i];

}

small = arr[0];

large = arr[4];

for (int i = 0;i < 5;i++)

{

if (arr[i] > large)

{

large = arr[i];

}

}

for (int i = 0;i < 5;i++)

{

if (arr[i] < small)

{

small = arr[i];

}

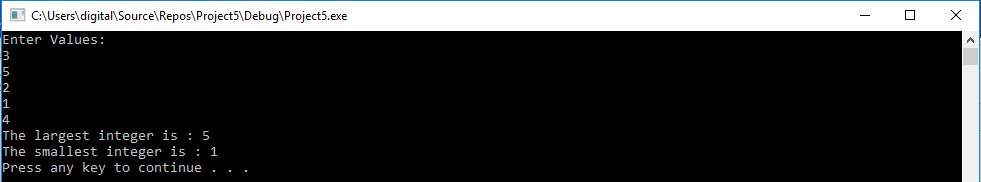
}

cout << "The largest integer is : " << large << endl;

cout << "The smallest integer is : " << small << endl;

system("pause");

}



**Q6:**

**Write a program to find the second largest element of an array.**

**Code:**

#include<iostream>

using namespace std;

int main()

{

int temp=0;

int arr[5];

cout << "Enter Values: " << endl;

for (int i = 0;i < 5;i++)

{

cin>> arr[i];

}

for (int i = 0;i < 5;i++)

{

for (int j = i + 1;j < 5;j++)

{

if (arr[i] > arr[j] )

{

temp=arr[j];

arr[j] = arr[i];

arr[i] = temp;

}

}

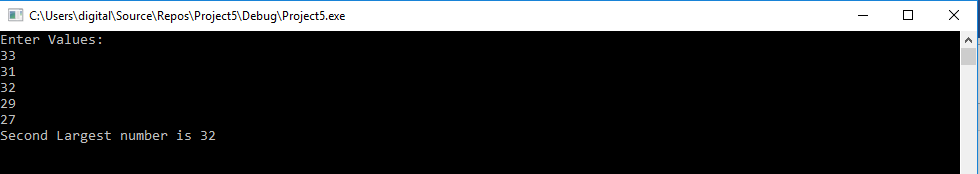
}

cout <<"Second Largest number is "<<arr[3]<<endl;

system("pause>0");

return 0;

}



**Q7:**

**Write a program that asks the user to type 10 integers of an array and an integer value**

**V. The program must search if the value V exists in the array and must remove the first**

**occurrence of V, shifting each following element left and adding a zero at the end of the**

**array. The program must then write the final array. All of this must be done using a**

**function to which the array and its size will be passed. Name the function Remove().**

**Code:**

#include<iostream>

using namespace std;

void remove(int arr[], int size, int v);

bool check(int arr[], int size, int v);

int main()

{

int arr[10], v;

cout << "Enter array numbers:";

for (int i = 0; i < 10; ++i)

{

cin >> arr[i];

}

cout << "Enter value:";

cin >> v;

remove(arr, 10, v);

cout << endl;

system("pause>0");

return 0;

}

void remove(int arr[], int size, int v)

{

int chec;

chec = check(arr, size, v);

if (chec == 1)

{

int x, y;

for (int i = 0; i < 10; ++i)

{

if (arr[i] == v)

{

y = i;

}

}

x = y;

for (int i = y; i < 10; i++)

{

if (i < 9)

arr[i] = arr[i + 1];

if (i == 9)

arr[i] = 0;

}

for (int i = 0; i < 10; i++)

{

cout << arr[i] << " ";

}

}

else

{

cout << "Invalid Value!" << endl;

}

}

bool check(int arr[], int size, int v)

{

for (int i = 0;i < size;i++)

{

if (arr[i] == v)

{

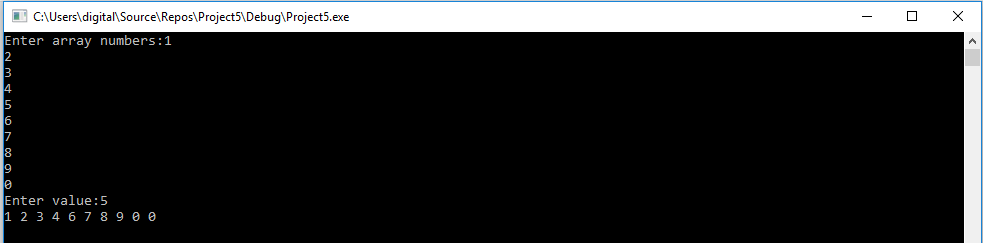
return 1;

}

}

return 0;

}



**Q8:**

Write a function which will be given as input an array, its size and an integer p. The

function will then cyclically shift the array p positions to the right: each element is moved

p positions to the right, while the last p elements are moved to the beginning of the

array. For example: if we have the array [ 1 2 3 4 5 6], shifting 2 positions to the right

should give the array [ 5 6 1 2 3 4 ]. Your function should work correctly for negative

values of p.

**Code:**

#include<iostream>

using namespace std;

void Pos(int arr[], int size, int p);

int main()

{

int arr[10], p;

cout << "Enter Six numbers: " << endl;

for (int i = 0; i < 6; ++i)

{

cin >> arr[i];

}

cout << "Enter position: ";

cin >> p;

cout << "After Shifting " << p << " times: ";

Pos(arr, 6, p);

cout << endl;

system("pause>0");

return 0;

}

void Pos(int arr[], int size, int p)

{

int temp;

if (p < 0)

{

p = (p\*-1);

}

for (int i = 0;i < p;i++)

{

temp = arr[i];

arr[i] = arr[size - i - 1];

arr[size - i - 1] = temp;

}

for (int i = 0;i < p;i++)

{

temp = arr[i+2];

arr[i+2] = arr[size - i - 1];

arr[size - i - 1] = temp;

}

for (int i = 0;i < 1;i++)

{

temp = arr[size-2];

arr[size-2] = arr[size- 1];

arr[size -1] = temp;

}

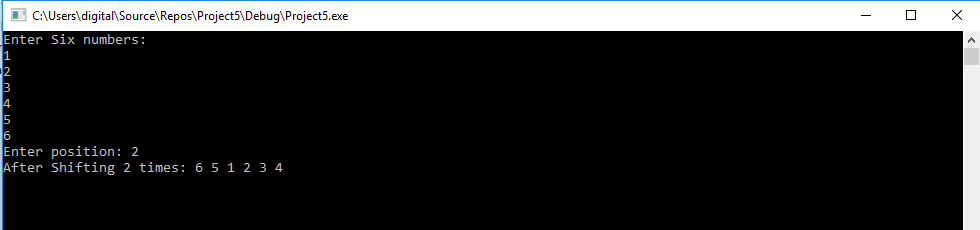
for (int i = 0;i < size;i++)

{

cout << arr[i] << " ";

}

}



**Q9:**

**Take 20 integer inputs from user and print the following:**

**number of positive numbers**

**number of negative numbers**

**number of odd numbers**

**number of even numbers**

**number of 0.**

**Code:**

#include<iostream>

using namespace std;

int main()

{

int even = 0, odd = 0, pos = 0, neg = 0, zero = 0, arr[20];

cout << "Enter 20 numbers : ";

for (int i = 0; i < 20; i++)

{

cin >> arr[i];

}

for (int i = 0; i < 20; i++)

{

if (arr[i] < 0)

{

neg++;

if (arr[i] % 2 == 0)

{

even++;

}

else

{

odd++;

}

}

else if (arr[i] == 0)

{

zero++;

}

else

{

pos++;

if (arr[i] % 2 == 0)

{

even++;

}

else

{

odd++;

}

}

}

cout << "even numbers = " << even << endl;

cout << "odd numbers = " << odd << endl;

cout << "Positive Numbers = " << pos << endl;

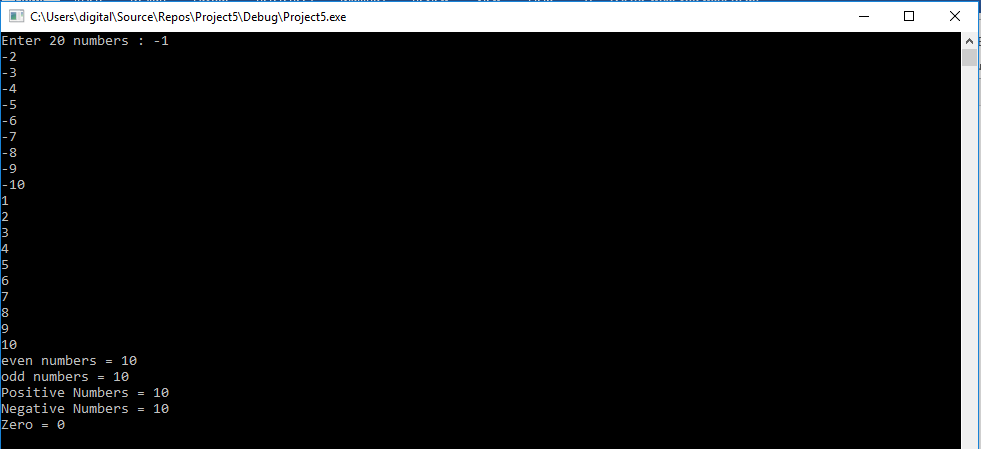
cout << "Negative Numbers = " << neg << endl;

cout << "Zero = " << zero << endl;

system("pause>0");

return 0;

}



**Q10:**

**Write a program to check if elements of an array are same or not it read from front or**

**back. E.g.-**

**2 3 15 15 3 2**

**Code:**

#include <iostream>

using namespace std;

void palindrome(int arr[], int size)

{

int count = 3;

for (int i = 0;i < size;i++)

{

cout << arr[i] << " ";

}

for (int i = 0; i < size; i++)

{

if (arr[i] != arr[size - 1 - i])

{

count++;

}

else

{

count--;

}

}

if (count>3)

{

cout << " Array is not Same." << endl;

}

else

{

cout << " Array is Same." << endl;

}

}

int main()

{

int arr[6], size = 6, check;

cout << "Enter Values: " << endl;

for (int i = 0; i < 6; i++)

{

cin >> arr[i];

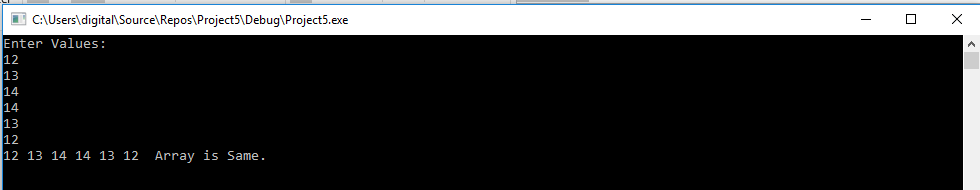
}

palindrome(arr, size);

system("pause>0");

return 0;

}



**Q11:**

**P is one-dimensional array of integers. Write a C++ function to efficiently search for a**

**data VAL from P. If VAL is present in the array then the function should return value 1**

**and 0 otherwise.**

**Code:**

#include <iostream>

using namespace std;

bool palindrome(int arr[], int size, int value);

int main()

{

int arr[10], size = 10, check,numb=0;

cout << "Enter Values: " << endl;

for (int i = 0; i < 10; i++)

{

cin >> arr[i];

}

cout << "Enter Number to check=";

cin >> numb;

check = palindrome(arr, size,numb);

if (check == 1)

cout << "Returned " << check << " So Value is present." << endl;

else

cout << "Returned " << check << " So Value is present." << endl;

system("pause>0");

return 0;

}

bool palindrome(int arr[], int size, int value)

{

for (int i = 0; i < size; i++)

{

if (arr[i] == value)

{

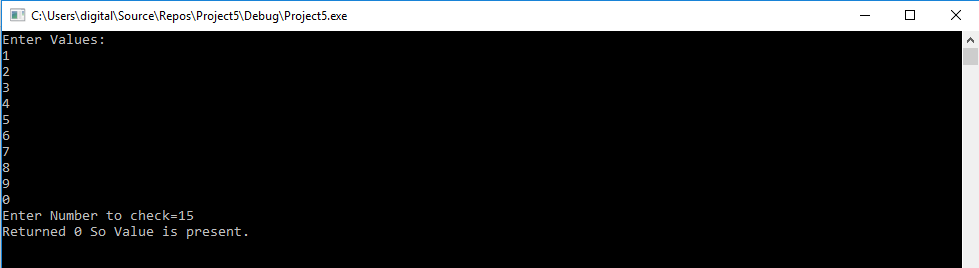
return 1;

}

}

return 0;

}



**Q12:**

**Given two arrays of integers A and B of sizes M and N respectively. Write a function**

**named MIX () with four arguments, which will produce a third array named C. such that**

**the following sequence is followed.**

**All even numbers of A from left to right are copied into C from left to right.**

**All odd numbers of A from left to right are copied into C from right to left.**

**All even numbers of B from left to right are copied into C from left to right.**

**All old numbers of B from left to right are copied into C from right to left.**

**A, B and C are passed as arguments to MIX (). e.g., A is {3, 2, 1, 7, 6, 3} and B is {9, 3,**

**5, 6, 2, 8, 10} the resultant array C is {2, 6, 6, 2, 8, 10, 5, 3, 9, 3, 7, 1, 3}**

**Code:**

#include<iostream>

using namespace std;

void MIX(int A[],int M, int B[],int N, int C[]);

int main()

{

int A[5], B[5], C[10], b = 9;

cout << "Enter the values in array A:";

for (int i = 0; i < 5; i++)

{

cin >> A[i];

}

cout << "Enter the values in array B:";

for (int i = 0; i < 5; i++)

{

cin >> B[i];

}

cout << "Array C is:";

MIX(A,5, B,5, C);

cout << endl;

system("pause>0");

return 0;

}

void MIX(int A[], int M, int B[], int N, int C[])

{

int a = 0,b=9;

for (int i = 0; i < 5; i++)

{

if (A[i] % 2 == 0)

{

C[a] = A[i];

a++;

}

else

{

C[b] = A[i];

b--;

}

}

for (int i = 0; i < 5; i++)

{

if (B[i] % 2 == 0)

{

C[a] = B[i];

a++;

}

else

{

C[b] = B[i];

b--;

}

}

for (int i = 0; i < 10; i++)

{

cout << C[i] << " ";

}

}

