**PF Lab no. 10**

**University of Central Punjab**

**Faculty of Information Technology**

**Spring 2023**

**Lecturer: Hafiz Bilal Shahid**

**Name: Muhammad Ahmad**

**Roll No.: L1F22BSCS0634**

**Section: B12**

* **Task no. 1: -**

**Program that takes 5 array elements from user. Swap position [2] element With position [4] element.**

#include<iostream>

using namespace std;

// Function to take input from user

void inputArray(int\* arr, int size\_f) {

cout << "Enter Elements: ";

for (int a = 0; a < size\_f; a++) {

cin >> arr[a];

}

}

// Function to print the array

void printArray(int\* arr, int size) {

cout << "Array Elements: ";

for (int a = 0; a < size; a++) {

cout << arr[a] << " ";

}

cout << endl;

}

// Function to swap the two values using a new dynamic array

int\* swap(int\* oldarray) {

int\* newarray = new int[5];

for (int a = 0; a < 5; a++) {

if (a == 2) {

newarray[a] = oldarray[4];

}

else if (a == 4) {

newarray[a] = oldarray[2];

}

else {

newarray[a] = oldarray[a];

}

}

delete[] oldarray;

oldarray = NULL;

return newarray;

}

int main()

{

// Declaring a Dynamic Array

int\* array = new int[5];

// Calling Functions

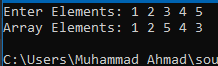
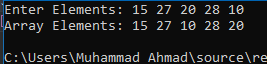
inputArray(array, 5);

array = swap(array);

printArray(array, 5);

return 0;

}

**Output:**

* **Task no. 2: -**

**program to input arrays values by using functions and pointer array. Make a function of name “inputArray” in which user enters the value of array and then display the values in main code:**

#include <iostream>

using namespace std;

// Function to take input from user

void inputArray(int\* arr, int size) {

cout << "Enter Elements: ";

for (int a = 0; a < size; a++) {

cin >> arr[a];

}

}

// Function to print the array

void printArray(int\* arr, int size) {

cout << "Array Elements: ";

for (int a = 0; a < size; a++) {

cout << arr[a] << " ";

}

cout << endl;

}

int main() {

int size;

// Taking input of size from user for dynamic memory allocation

cout << "Enter Size of Input: ";

cin >> size;

// Dynamically alloacting array

int\* arr = new int[size];

// Taking input from user through inputArray function

inputArray(arr, size);

printArray(arr, size);

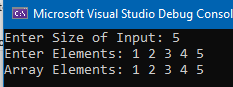
// Deallocating memory of array

delete[] arr;

arr = NULL;

return 0;

}

**Output:**

* **Task no. 3: -**

**Program to take array elements from user and then double them by multiplying them by 2 and then display the result:**

#include <iostream>

using namespace std;

// Function to take input from user

void inputArray(int\* arr, int size) {

cout << "Enter Elements: ";

for (int a = 0; a < size; a++) {

cin >> arr[a];

}

}

// Function to print array on console

void printArray(int\* arr, int size) {

cout << "Array Elements: ";

for (int a = 0; a < size; a++) {

cout << arr[a] << " ";

}

cout << endl;

}

// Funtion to double the each element of array

void doubarray(int\*arr,int size\_f) {

for (int a = 0; a < size\_f; a++) {

arr[a] \*= 2;

}

}

int main() {

int size;

cout << "Enter Size of Input: ";

cin >> size;

int\* array = new int[size];

inputArray(array, size);

cout << "Elements in Array: ";

printArray(array, size);

doubarray(array, size);

cout << "Elements in Array after Double: ";

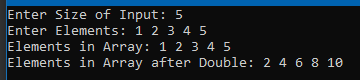
printArray(array, size);

delete[] array;

array = NULL;

return 0;

}

**Output: -**

* **Task no. 4: -**

**Program for ATM cash withdrawal that utilizes dynamic memory allocation, arrays, and handles the issue of dangling pointers:**

#include<iostream>

using namespace std;

// bool Function to chec if card is inserted or not

bool insertcard() {

int temp;

cout << "Please Insert Your Card" << endl;

cout << "Press 1 if you have inserted your card: ";

cin >> temp;

if (temp == 1) {

return temp;

}

else {

return 0;

}

}

int main()

{

int check = 0;

char pass[5] = { '1','2','3','4' };

char input\_pin[4];

int withdraw;

int\* balance = new int;

cout << "Enter Balance of Account: ";

cin >> \*balance;

int\* newbalance = new int;

cout << "\t\tWelcome To Meezan Bank ATM\n" << endl;

bool card = insertcard();

if (card == 1) {

cout << "\nEnter Your Password: ";

for (int a = 0; a < 4; a++) {

cin >> input\_pin[a];

if (input\_pin[a] == pass[a]) {

check++;

}

}

while (check != 4) {

check = 0;

cout << "\nWrong Password!" << endl;

cout << "Enter Your Password Again: ";

for (int a = 0; a < 4; a++) {

cin >> input\_pin[a];

if (input\_pin[a] == pass[a]) {

check++;

}

}

}

if (check == 4) {

cout << "Enter the Amount: ";

cin >> withdraw;

if (withdraw <= \*balance) {

\*newbalance = \*balance;

delete balance;

\*newbalance -= withdraw;

cout << "\nTransaction is Succesfull" << endl;

cout << "\nYour new Balance is: " << \*newbalance<< endl;

}

}

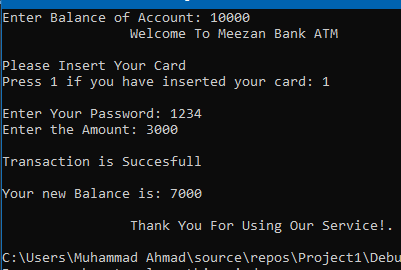
}

cout << "\n\t\tThank You For Using Our Service!." << endl;

delete newbalance;

return 0;

}

**Output:**

* **Task no. 5: -**

**program that takes 2 arrays of user’s choice. Find and display the Sum of elements of both arrays using Dynamic Memory Allocation:**

#include<iostream>

using namespace std;

// Function for taking input from user

void inputArray(int\* arr, int size\_f) {

for (int a = 0; a < size\_f; a++) {

cin >> arr[a];

}

}

// Function for printing array

void printArray(int\* arr, int size) {

cout << "\nResultant Array: ";

for (int a = 0; a < size; a++) {

cout << arr[a] << " ";

}

cout << endl;

}

// Funtion to calculate the resultant sum array of two arrays

int\* sum(int\* array1, int\* array2, int size\_f) {

int\* resultant = new int[size\_f];

for (int a = 0; a < size\_f; a++) {

resultant[a] = array1[a] + array2[a];

}

return resultant;

}

int main()

{

int size;

// Takinag size from user

cout << "Enter the size of Array: ";

cin >> size;

// Dynamically allocating arrays according to user input

int\* arr1 = new int[size];

int\* arr2 = new int[size];

int\* result = NULL;

// Taking input through functions

cout << "\nEnter 1st Array Elements: ";

inputArray(arr1, size);

cout << "\nEnter 2nd Array Elements: ";

inputArray(arr2, size);

// Calling the sum function to calculate the sum of two arrays

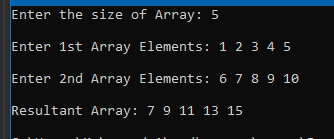
result = sum(arr1, arr2, size);

// Printing resultant array using printArray function

printArray(result, size);

return 0;

}

**Output: -**

* **Task no. 6: -**

**program that takes roll number and marks of students from user in an array. Display the roll number and marks of each student using Dynamic Memory Allocation:**

#include<iostream>

using namespace std;

// Function for taking input of roll number and marks from user

void input(int\* roll\_f,int\*marks\_f, int size\_f) {

for (int a = 0; a < size\_f; a++) {

cout << "Enter the Roll Number of Student " << a + 1 << ": ";

cin >> roll\_f[a];

cout << "Enter the Marks of Student " << a + 1 << ": ";

cin >> marks\_f[a];

cout << endl;

}

}

// Function for printing the roll no. and marks

void printArray(int\* roll\_f,int\* marks\_f, int size\_f) {

cout << "Roll\t\tMarks" << endl;

for (int a = 0; a < size\_f; a++) {

cout << roll\_f[a] << "\t\t" << marks\_f[a] << endl;

}

cout << endl;

}

int main()

{

int size;

// Taking Size of class from user

cout << "Enter Total Number of Students: ";

cin >> size;

cout << endl;

// Dynamically Allocating memory

int\* roll = new int[size];

int\* marks= new int[size];

input(roll,marks, size);

printArray(roll,marks, size);

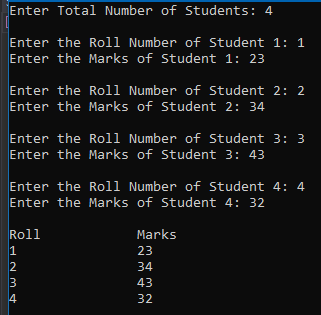
// Deallocating the arrays

delete[] roll;

delete[] marks;

return 0;

}

**Output: -**

* **Task no. 7: -**

**Program to output if a number is prime or not by using dynamic variable. If the number is prime, then find its factorial. Complete this task by using infinite while loop:**

#include<iostream>

using namespace std;

// Boolean returnig function to decide either number is prime or not

bool findprime(int num) {

int check = 0;

for (int a = 1; a <= num; a++) {

if (num % a == 0) {

check++;

}

}

if (check == 2) {

return true;

}

else {

return false;

}

}

// Function to print factorial od given number

void factorial(int num) {

int answer=1;

cout << "Factorial: " << endl;

for (int a = 1; a <= num; a++) {

if (a == num) {

cout << a << " = ";

}

else {

cout << a << " x ";

}

answer \*= a;

}

cout << answer << endl;

}

int main()

{

cout << "\t\tPrime Number Finder" << endl;

// infinite while loop

while (true) {

int\* input = NULL;

input = new int;

// Taking input of integer from user

cout << "\nEnter the Number: ";

cin >> \*input;

// Condition to print the factoral and result

if (findprime(\*input) == 1) {

cout << "Number " << \*input << " is Prime." << endl;

factorial(\*input);

}

else {

cout<<"Number " << \*input << " is not a Prime." << endl;

}

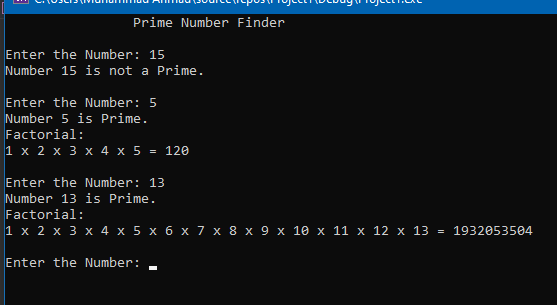
// Deallocate the input memory

delete input;

}

return 0;

}

**Output: -**