Федеральное государственное автономное образовательное учреждение

Санкт-Петербургский политехнический университет Петра Великого

(ФГАОУ ВО «СПБПУ»)

Институт среднего профессионального образования

ОТЧЕТ

Лабораторная работа №2 Вариант 7

О выполнении задачи

по дисциплине: «Основы программирования»

Выполнил студент: Балахонский ДМ

группы 22928/3

Преподаватель: Уцеховский А.А.

Задание:

- создать динамический одномерный массив M01 длиной L элементов (длина массива – см. п.1 варианта);

- заполнить массив целыми случайными числами из интервала (см. п.2 варианта);

- массив M01 вывести на консоль (вывод в одну строку с разделителями), для этого написать функцию и использовать ее в дальнейшем;

- создать новый массив M02 на основе массива M01, для этого написать функцию; выполнить над элементами массива M02 арифметические операции (см. п.3 варианта), результат вывести на консоль, используя функцию;

- выполнить циклическую перестановку элементов массива M02, для этого написать отдельную функцию, (направление перестановки, вправо или влево, и число позиций перестановки - см. п.4 варианта), результат вывести на консоль;

- отсортировать массив M02 методом пузырька по возрастанию или убыванию (см. п.5 варианта), результат вывести на консоль;

- создать новый массив M03 той же длины L и заполнить его нулями; переписать в него элементы массива M02 случайным образом (последовательно перебираем элементы массива M02, индекс элемента массива M03 генерируем случайным образом и переписываем в него элемент из M02, если же элемент массива M03 занят – ненулевой, продолжаем генерировать случайные индексы), результат вывести на консоль.

* удалить все созданные массивы;

Программу отработать 4 раза при условии, что каждый раз генерируется новая выборка случайных чисел.

Код:

#include <iostream>  
#include <vector>  
#include <cmath>  
#include <ctime>  
  
using namespace std;  
  
const int arraySizeFrom = 7;  
const int arraySizeTo = 18;  
const int arrayRangeFrom = 4;  
const int arrayRangeTo = 11;  
const int arrayShiftFrom = 4;  
const int arrayShiftTo = 6;  
  
int random(int from, int to){  
 return rand() % (to - from + 1) + from;  
}  
  
const void arrayOutput(vector<int> arr){  
 for(auto e : arr){  
 cout << e << " | ";  
 }  
 cout << endl;  
}  
  
vector<int> arrayShiftLeft(vector<int> arr){  
 int temp = arr[0];  
 for(int i = 0; i < arr.size(); i++){  
 arr[i] = arr[i + 1];  
 }  
 arr[arr.size() - 1] = temp;  
 return arr;  
}  
  
vector<int> loopedArrayShiftLeft(vector<int> arr, int shiftingNumber){  
 vector<int> test;  
 for(int i = 0; i < shiftingNumber; i++){  
 if (i == 0){  
 test = arrayShiftLeft(arr);  
 }else {  
 test = arrayShiftLeft(test);  
 }  
 }  
 return test;  
}  
  
vector<int> arrayShiftRight(vector<int> arr){  
 int temp = arr[arr.size() - 1];  
 for(int i = arr.size() - 2; i >= 0; i--){  
 arr[i + 1] = arr[i];  
 }  
 arr[0] = temp;  
 return arr;  
}  
  
vector<int> loopedArrayShiftRight(vector<int> arr, int shiftingNumber){  
 vector<int> test;  
 for(int i = 0; i < shiftingNumber; i++){  
 if (i == 0){  
 test = arrayShiftRight(arr);  
 }else {  
 test = arrayShiftRight(test);  
 }  
 }  
 return test;  
}  
  
void bubbleSort(vector<int>& a, bool des = false){  
 bool swapp = true;  
 while(swapp){  
 swapp = false;  
 for (size\_t i = 0; i < a.size() - 1; i++) {  
  
 if (des){  
 if (a[i] < a[i + 1] ){  
 a[i] += a[i + 1];  
 a[i + 1] = a[i] - a[i + 1];  
 a[i] -= a[i + 1];  
 swapp = true;  
 }  
 }else{  
 if (a[i] > a[i + 1] ){  
 a[i] += a[i + 1];  
 a[i + 1] = a[i] - a[i + 1];  
 a[i] -= a[i + 1];  
 swapp = true;  
 }  
 }  
 }  
 }  
}  
  
int main(){  
 // setting token for rand number  
 srand(time(NULL));  
 const int L = random(arraySizeFrom, arraySizeTo);  
 vector<int> M01;  
  
 //filling the array  
 for(int i = 0; i < L; i++){  
 M01.push\_back(random(arrayRangeFrom, arrayRangeTo));  
 }  
  
 cout << "Array length: " << L << endl;  
 cout << "\nGenerated array M01[" << L << "]:" << endl;  
 arrayOutput(M01);  
  
 vector<int> M02 = M01;  
  
 for(int i = 0; i < L; i++){  
 if (M02[i] <= 6) {  
 M02[i] = M02[i] + 3 ;  
 }  
 }  
  
 cout << "\nAdd 3 to array elements that are less than or equal to 6" << endl; // Ко всем элементам массива M02 , которые <= 6, прибавить 3;  
 arrayOutput(M02);  
  
 const int arrayShiftIndex = random(arrayShiftFrom, arrayShiftTo);  
 if(M02[1] % 2 == 1){  
 cout << "\nThe array was shifted left for "<< arrayShiftIndex <<" positions" << endl;  
 arrayOutput(loopedArrayShiftLeft(M02, arrayShiftIndex));  
 }else{  
 cout << "\nThe array was shifted right for "<< arrayShiftIndex <<" positions" << endl;  
 arrayOutput(loopedArrayShiftRight(M02, arrayShiftIndex));  
 }  
  
 bubbleSort(M02, true);  
  
 cout << "\nThe array was bubble sorted in descending order" << endl;  
 arrayOutput(M02);  
  
 vector<int> M03 = {};  
 for(int i = 0; i < L; i++){  
 M03.push\_back(0);  
 }  
  
 vector<int> M02C = M02;  
  
 for(int i = 0; i < M02.size(); i++){  
 int rIndex = random(0, M02C.size() - 1);  
 M03[i] = M02C[rIndex];  
 M02C.erase(M02C.begin() + rIndex);  
 }  
  
 cout << "\nThe array M03 was created randomly with M02's elements" << endl;  
 arrayOutput(M03);  
}

Результаты выполнения программы:





