**Лабораторна робота №2**

**Розробка та реалізація програм для роботи з вказівниками та одновимірними динамічними масивами**

**Ціль роботи**: оволодіння навичками складання програм для роботи з вказівниками та одновимірними динамічними масивами.

**Завдання**

**Завдання 2.1.** Нехай є наступний фрагмент програми. Поясніть, як зміниться масив після його виконання, при наступному початку програми.

#include <iostream>

#include <cmath>

using namespace std;

int\* form(int &n) {

n = 10 + n % 10;

int \*a = new int[n];

\*a = n;

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

\*(a + i) = \*(a + i - 1) + 1;

}

return a;

}

*Розв’язання*

#include <iostream>

#include <cmath>

using namespace std;

//Генерирует массив чисел

int\* form(int &n) {

n = 10 + n % 10; //10 + последняя цыфра варианта (менят переданную переменную)

int \*a = new int[n];

\*a = n; //a[0] = n -> 15

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

\*(a + i) = \*(a + i - 1) + 1;

//a[i] = a[i - 1] + 1;

}

return a;

}

int main() {

int n = 5;

int \*a = form(n);

//a: 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29

int \*\*p = new int \*[n];

for (int i = 0; i < n; i++) {

p[i] = (a + i); //часть массива а начиная с i

\*p[i] = \*p[i] + n; //увеличение a[i] на n

}

//каждый элемент увеличен на n (на 15)

//a: 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44

. . .

**Завдання 2.2.**

*Розв’язання*

1. Постановка задачі:

Cклаcти програму, що заповнює динамічний одновимірний масив за допомогою випадкових чисел, а потім видаляє усі всі парні елементи

1. Алгоритм розв’язання задачі:

Алгоритм розв’язання задачі можна представити у вигляді такої послідовності дій:

Дія 1. Виділити динамічну пам'ять для елементів масиву.

Дія 2. Заповнити масив випадковими числами.

Дія 3. Вивести елементи масиву на екран.

Дія 4. Визначити кількість елементів нового масиву.

Дія 5. Виділити динамічну пам'ять для нового масиву.

Дія 6. У новий масив записати тільки парні елементи.

Дія 7. Звільнити пам'ять, що була виділена під старий масив.

Дія 8. Замінити значення вказівки.

Дія 9. Вивесит елементи масиву на екран.

1. **Текст програми**

#include <iostream>

#include <fstream>

#include <cmath>

#include <ctime>

using namespace std;

int\* removePairElements(int \*arr, int &len);

void writeArr(int \*arr, int len);

void printArr(int \*arr, int len, ofstream &fout, char sep = ' ');

int getCountPairElements(int \*arr, int len);

int main() {

ifstream fin("input.txt");

ofstream fout("output.txt");

srand(time(0));

int len;

fin >> len;

fout << "Длина массива: " << len << endl;

int \*arr = new int[len];

writeArr(arr, len);

fout << endl << "Был сгенерирован массив: " << endl;

printArr(arr, len, fout, ' ');

arr = removePairElements(arr, len);

fout << endl << "После удаления: " << endl;

printArr(arr, len, fout, ' ');

fin.close();

fout.close();

if (arr) delete[] arr;

return 0;

}

int\* removePairElements(int \*arr, int &len) {

if (!arr) return arr;

int newLen = getCountPairElements(arr, len);

int \*res = new int[newLen];

int size = -1;

for (int i = 0; i < len; i++) {

if (\*(arr + i) % 2 == 0) {

\*(res + ++size) = \*(arr + i);

}

}

len = newLen;

delete[] arr;

return res;

}

int getCountPairElements(int \*arr, int len) {

int count = 0;

for (int i = 0; i < len; i++) {

if ( \*(arr + i) % 2 == 0 ) count++;

}

return count;

}

void writeArr(int \*arr, int len) {

for (int i = 0; i < len; i++) {

\*(arr + i) = 10 + rand() % 90;

}

}

void printArr(int \*arr, int len, ofstream &fout, char sep) {

for (int i = 0; i < len; i++) {

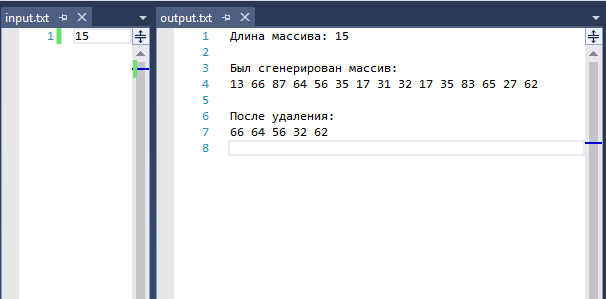
fout << \*(arr + i) << sep;

}

fout << endl;

}

1. Результат виконання програми:



**Висновок:** на цій лабораторній роботі я оволодів навичками складання програм для роботи з вказівниками та одновимірними динамічними масивами.