// ConsoleApplication12.cpp: определяет точку входа для консольного приложения.

//

#include "stdafx.h"

#include <iostream>

#include <fstream>

#include <string>

using namespace std;

\_\_interface IFileOperation

{

void saveTofile(string fileName);

void readFromFile(string fileName);

};

//---------------

class Array :public IFileOperation

{

private:

double\* data;

int MaxLength;

int Length;

void setMaxLength(int value)

{

if (value <= 0) throw "Error. Length<=0";

data = new double[value];

Length = 0;

MaxLength = value;

}

void extendArray() //Збільшуємо місткість масиву вдвічі

{

double\* temp = new double[MaxLength\*2]; //Створюмєо вдвічі більший масив

for (int i = 0; i < MaxLength; i++) //Копіюємо значення у новий масив

{

temp[i] = data[i];

}

delete[] data; //Старий масив видаляємо

data = temp; //У data записуємо адресу нового масиву

MaxLength \*= 2;

}

bool isIndexCorrect(int index)

{

return index >= 0 && index < Length;

}

public:

int getLength()

{

return Length;

}

void setElement(int index, double value)

{

if (!isIndexCorrect(index)) throw "Out of Range";

data[index] = value;

}

double getElement(int index)

{

if(!isIndexCorrect(index)) throw "Out of Range";

return data[index];

}

Array(int count, double first, ...)

{

setMaxLength(count);

double\* arr = &first;

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

{

data[i] = arr[i];

}

Length = count;

}

Array(string fileName)

{

readFromFile(fileName);

}

Array(const Array& otherArray)

{

setMaxLength(otherArray.Length);

for (int i = 0; i < otherArray.Length; i++)

{

data[i] = otherArray.data[i];

}

Length = otherArray.Length;

}

Array()

{

setMaxLength(20);

Length = 0;

}

void print()

{

printf("Array: %s\n",toString().data());

}

string toString()

{

if (Length == 0)

return "Array is empty.";

string s = "[" + to\_string(data[0]); //[12, 4, 23, 78]

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

{

s += ", " + to\_string(data[i]);

}

s += "]";

return s;

}

void input()

{

delete[] data; //Видаляємо старі елементи

int count; //Вводимо нову кількість елементів

printf("count = ");

cin >> count;

setMaxLength(count); //Встановлюємо максимальну двожину масиву

for (int i = 0; i < count; i++) //Вводимо елементи з клавіатури

{

printf("Element %d : ",i);

cin >> data[i];

}

Length = count; //Запам"ятовуємо поточну кількість елементів

}

void addElement(double value)

{

if (Length == MaxLength)

extendArray();

data[Length] = value;

Length++;

}

void deleteElementByIndex(int index)

{

if (!isIndexCorrect(index)) throw "Index out of Range";

for (int i = index+1; i < Length; i++)

{

data[i - 1] = data[i];

}

Length--;

}

int indexOf(double value)

{

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

{

if ( abs(data[i] - value)<0.0000001)

return i;

}

return -1;

}

bool remove(double value)

{

int index = indexOf(value);

if (index == -1)

return false;

deleteElementByIndex(index);

return true;

}

int removeAll(double value)

{

int count = 0;

while (remove(value)) count++;

return count;

}

void saveTofile(string fileName)

{

ofstream file(fileName); //Відкриваємо файл для запису

//---- Запис даних у файл (файл << дані)

file << MaxLength << endl; //Записуємо максимальну кількість

file << Length << endl; //Записуємо поточну кількість

for (int i = 0; i < Length; i++) //Записуємо через пробіл елементи

{

file << data[i] << " ";

}

//file.write((char \*)data, sizeof(double)\*Length); //file.write((char\*) адреса\_поч.\_даних, кільк.байтів)

file.close(); //Закриваємо файл

}

void readFromFile(string fileName)

{

delete[] data;

ifstream file(fileName); //Відкриваємо файл для читання

int temp;

file >> temp; //Зчитуємо значення максимальної кількості

setMaxLength(temp);

file >> temp; //Зчитуємо значення справжньої кількості

Length = temp;

for (int i = 0; i < Length; i++) //Зчитуємо значення елементів

{

file >> data[i];

}

file.close(); //Закриваємо файл

}

//------- Перевантаження операцій

double &operator[](int index) {

if (!isIndexCorrect(index)) throw "Index out of range";

return data[index];

}

~Array()

{

delete[] data;

}

};

int main()

{

Array\* arr = new Array();

while (true)

{

printf("1.Add element\n");

printf("2.Remove element by index\n");

printf("3.Print Array\n");

printf("4.Save to file\n");

printf("5.Read from file\n");

printf("6.Exit\n");

printf("Answer: ");

int ans;

cin >> ans;

string fileName;

switch (ans)

{

case 1:printf("Element = ");

double value;

cin >> value;

arr->addElement(value);

break;

case 2:printf("Insdex = ");

int index;

cin >> index;

arr->deleteElementByIndex(index);

break;

case 3:

arr->print();

break;

case 4:

printf("File name : ");

cin >> fileName;

arr->saveTofile(fileName);

break;

case 5:

printf("File name : ");

cin >> fileName;

arr->readFromFile(fileName);

break;

case 6:return 0;

}

}

return 0;

}