#include <iostream>

#include <string>

#include <Windows.h>

using namespace std;

class String {

private:

string m\_array[10];

public:

String() {}; //конструктор

String(const String&ob); //конструктор копирования

string& operator[] (const int index); //перегрузка индексирования

string& operatorplus(const string plus, const int index); //перегрузка сложения

friend bool operator> (const String &s1, const String &s2);

friend bool operator<= (const String &s1, const String &s2);

friend bool operator< (const String &s1, const String &s2);

friend bool operator>= (const String &s1, const String &s2);

void Konkatenacia(string s1, string s2);

void Izvlechenie(string s1, int nomer);

void Sravnenie(string s1, string s2);

~String() { delete m\_array; };//деструктор

};

bool operator> (const String &s1, const String &s2)

{

return s1.m\_array > s2.m\_array;

}

bool operator>= (const String &s1, const String &s2)

{

return s1.m\_array >= s2.m\_array;

}

bool operator< (const String &s1, const String &s2)

{

return s1.m\_array < s2.m\_array;

}

bool operator<= (const String &s1, const String &s2)

{

return s1.m\_array <= s2.m\_array;

}

string& String::operatorplus(const string plus, const int index)

{

m\_array[index] += plus;

return m\_array[index];

}

string& String::operator[] (const int index)

{

return m\_array[index];

}

void String::Konkatenacia(string s1, string s2) {

cout << s1 + s2;

}

void String::Izvlechenie(string s1, int nomer) {

char sim = s1[nomer - 1];

cout << sim;

}

void String::Sravnenie(string s1, string s2) {

if (s1 == s2) cout << "Строки одинаковые: " << s1 << " = " << s2 << endl;

else cout << "Строки не одинаковые: " << s1 << " != " << s2 << endl;

if (s1.length() == s2.length()) cout << "Строки одинаковой длины: " << s1.length() << endl;

else if (s1.length() > s2.length()) cout << "Строки не одинаковой длины, s1>s2: " << s1.length() << " > " << s2.length() << endl;

else if (s1.length() < s2.length()) cout << "Строки не одинаковой длины, s1<s2: " << s1.length() << " < " << s2.length() << endl;

}

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

String s;

String arrays;

string mas[10];

setlocale(LC\_ALL, "Rus");

int chis;

string s1, s2;

cout << "Введите строки: " << endl << "s1: " << endl;

cin >> s1;

cout << "s2: " << endl;

cin >> s2;

/////////////////////////////////////////////

cout << endl << "Конкатенация: ";

cout << endl;

s.Konkatenacia(s1, s2);

cout << endl;

/////////////////////////////////////////////

cout << endl << "Вывод выбранных по счету символов: ";

cout << endl << "Введите какой по счету символ строки s1 хотите вывести: ";

cin >> chis;

s.Izvlechenie(s1, chis);

cout << endl << "Введите какой по счету символ строки s2 хотите вывести: ";

cin >> chis;

s.Izvlechenie(s2, chis);

cout << endl;

/////////////////////////////////////////////

cout << endl << "Сравнение строк: ";

s.Sravnenie(s1, s2);

cout << endl;

/////////////////////////////////////////////

cout << endl << "Введите массив для сортировки (10 элементов)" << endl;

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

cin >> arrays[i];

}

for (int i = 0; i < 9; i++)

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

if (strcmp(arrays[i].c\_str(), arrays[j].c\_str())>0) swap(arrays[i], arrays[j]);

}

cout << endl << "Отсортированный массив: " << endl;

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

cout << arrays[i] << endl;

}

system("pause");

return 0;

}



