Беларускі дзяржаўны тэхналагічны ўніверсітэт

Факультэт інфармацыйных тэхналогій

Кафедра праграмнай інжэнерыі

 Лабараторная работа 16

Па дысцыпліне «Асновы алгарытмізацыі і праграмавання»

На тэму «Функцыі карастальніка»

Выканала:

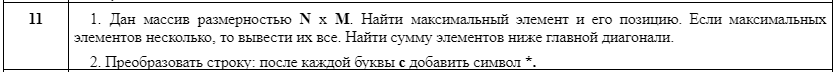
Студэнтка 1 курса 6 группы

Лускіна Вольга Аляксандраўна

Выкладчык: асс. Андронава М.В.

2023, Мінск

Варыянт 11



Заданне 1, 2

#include <iostream>

#include <string>

#include <Windows.h>

using namespace std;

void poshuk\_max(int\*\* arr, int n, int m) // Функцыя для пошуку найбольшага элемента ў матрыцы

{

int max = arr[0][0]; // Ініцыалізіруем зменную max значэннем першага элемента матрыцы

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

{

for (int j = 0; j < m; j++)

{

if (max < arr[i][j]) // Калі бягучы элемент большы за max, абнаўляем

{

max = arr[i][j];

}

}

}

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

{

for (int j = 0; j < m; j++)

{

if (max == arr[i][j])

{

cout << i + 1 << " " << j + 1 << ", "; // Выводзім каардынаты

}

}

}

}

int padlick\_sum(int\*\* arr, int n, int m) // Функцыя для падліку сумы элементаў матрыцы, якія стаяць пад галоўнай дыяганалью

{

int sum = 0;

for (int i = 1; i < n; i++) // Ідзём па радках матрыцы, пачынаючы з другога

{

for (int j = 0; j < i; j++) // Ідзём па слупках матрыцы да бягучага радка

{

sum += arr[i][j]; // дадаем значэнне бягучага элемента да сумы

}

}

return sum;

}

void poshuck\_s(string s) // Функцыя, каб пасля кожнай "с" у радку ставілася "\*"

{

for (int i = 0; i < s.length(); i++) // Ідзём па ўсім сімвалам радка

{

cout << s[i]; // Выводзім бягучы сімвал

if (s[i] == 'с') // Калі гэты сімвал "с", дадаем да яго \*

{

cout << "\*";

}

}

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

char exitWord[10] = "кандэлябр"; // Калі будзе ўведзена гэтае слова, праграма скончыцца

char userInput[10];

do {

int choice = 0;

bool validChoice = false; // Ставім сцяг

do {

cout << "Выбярыце заданне (1 альбо 2): ";

cin >> choice;

if (cin.fail() || choice < 1 || choice > 2) { // Праверка на правільнасць увода

cin.clear();

cin.ignore(1000, '\n');

cout << "Памылка. Калі ласка, выбярыце нумар задання ад 1 да 2: " << endl;

}

else {

validChoice = true;

}

}

while (!validChoice);

cin.clear();

switch (choice)

{

case 1:

{

int\*\* arr;

int n, m;

cout << "Увядзіце памер масіва: ";

cin >> n >> m;

arr = new int\* [n];

cout << "Увядзіце элементы масіва:" << endl;

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

{

arr[i] = new int[m];

}

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

for (int j = 0; j < m; j++)

cin >> arr[i][j];

cout << "Масіў:" << endl;

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

{

for (int j = 0; j < m; j++)

{

cout << arr[i][j] << " ";

}

cout << endl;

}

cout << "Каардыната максімальнага элемента: ";

poshuk\_max(arr, n, m);

cout << endl;

cout << "Сума элементаў пад галоўнай дыяганалью: " << padlick\_sum(arr, n, m) << endl;

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

{

delete[] arr[i];

}

delete[] arr; // Ачышчаем памяць, якую выдзялілі для двумернага масіва

break;

}

case 2:

{

string s;

cout << "Увядзіце радок: ";

cin.ignore();

getline(cin, s);

cout << "Новы радок: ";

poshuck\_s(s);

cout << endl;

break;

}

}

cout << "Увядзіце слова \"" << exitWord << "\", калі хочаце скончыць праграму, калі не, што заўгодна іншае: ";

cin.ignore();

cin.getline(userInput, sizeof(userInput)); // Чытаем радок карыстальнік

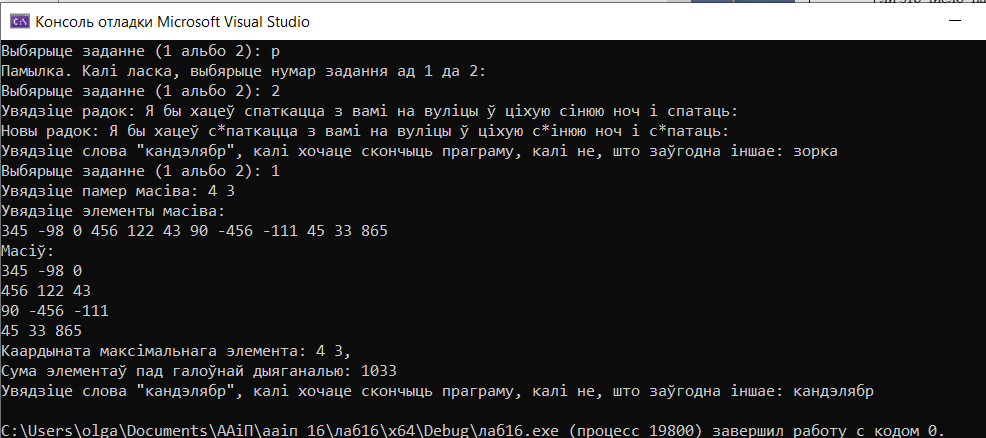
if (strcmp(userInput, exitWord) == 0) { // Параўноваем то, што было ўведзена, з тым, што патрабуецца для таго, каб скончыць праграму

break;

}

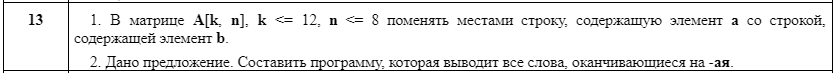
} while (true); // Бясконцы цыкл

}



Дадатковыя заданні:

Варыянт 13



Заданне 1, 2:

#include <iostream>

#include <string> // Бібліятэка для задання з сказам

#include <Windows.h>

using namespace std;

void ZmenaRadkouMesczami(int\*\* matrix, int k, int n, int a, int b) {

int aRad, bRad;

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

{

for (int j = 0; j < n; j++) // Пошук радкоў з элементамі а і b

{

if (matrix[i][j] == a)

{

aRad = i;

}

if (matrix[i][j] == b)

{

bRad = i;

}

}

}

for (int j = 0; j < n; j++) // Мяняем радкі месцамі

{

int temp = matrix[aRad][j];

matrix[aRad][j] = matrix[bRad][j];

matrix[bRad][j] = temp;

}

cout << "Новая матрыца: " << endl;

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

{

for (int j = 0; j < n; j++)

{

cout << matrix[i][j] << " ";

}

cout << "\n";

}

}

void SloviAya(const string& skaz) {

string word;

for (int i = 0; i <= skaz.length(); i++) {

char c = skaz[i];

if (c == ' ' || c == '.' || c == ',' || i == skaz.length()) {

if (!word.empty() && word.length() >= 2 && word.substr(word.length() - 2) == "ая") { // Глядзім, каб даўжыня слова была больш за 2 літары і дзве апошгія літары былі якраз "ая"

cout << word << endl;

}

word.clear();

}

else {

word.push\_back(c);

}

}

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

char exitWord[10] = "дурнязаўр"; // Калі будзе ўведзена гэтае слова, праграма скончыцца

char userInput[10];

do {

int choice = 0;

bool validChoice = false; // Ставім сцяг

do {

cout << "Выбярыце заданне (1 альбо 2): ";

cin >> choice;

if (cin.fail() || choice < 1 || choice > 2) { // Праверка на правільнасць увода

cin.clear();

cin.ignore(1000, '\n');

cout << "Памылка. Калі ласка, выбярыце нумар задання ад 1 да 2: " << endl;

}

else {

validChoice = true;

}

}

while (!validChoice);

cin.clear();

switch (choice)

{

case 1:

{

int k, n;

cout << "Увядзіце памер матрыцы(k <=12 i n<=8): ";

cin >> k >> n;

if (k <= 12 and n <= 8 and n > 0 and k > 0) // Праверка ўводу

{

int\*\* A = new int\* [k];

cout << "Увядзіце элементы масіву: " << endl;

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

{

A[i] = new int[n];

for (int j = 0; j < n; j++)

{

cin >> A[i][j];

}

}

cout << "Масіў:" << endl;

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

{

for (int j = 0; j < n; j++)

{

cout << A[i][j] << " ";

}

cout << endl;

}

cout << "Увядзіце 2 значэння, радкі з якімі трэба памяняць месцамі: ";

int a, b;

cin >> a >> b;

ZmenaRadkouMesczami(A, k, n, a, b);

}

break;

}

case 2:

{

string sentence;

cout << "Увядзіце сказ: ";

cin.ignore();

getline(cin, sentence);

cout << "Словы, у якіх апошнія літары -ая:" << endl;

SloviAya(sentence);

break;

}

}

cout << "Увядзіце слова \"" << exitWord << "\", калі хочаце скончыць праграму, калі не, што заўгодна іншае: ";

cin.ignore();

cin.getline(userInput, sizeof(userInput)); // Чытаем радок карыстальніка

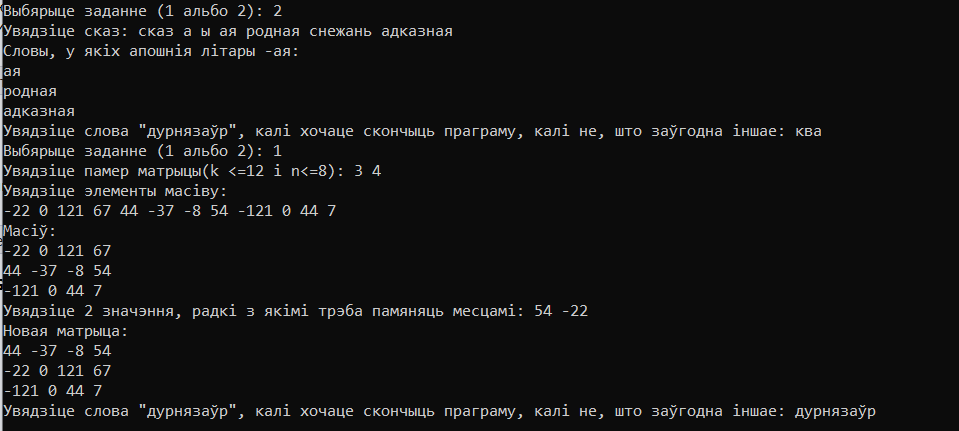
if (strcmp(userInput, exitWord) == 0) { // Параўноваем то, што было ўведзена, з тым, што патрабуецца для таго, каб скончыць праграму

break;

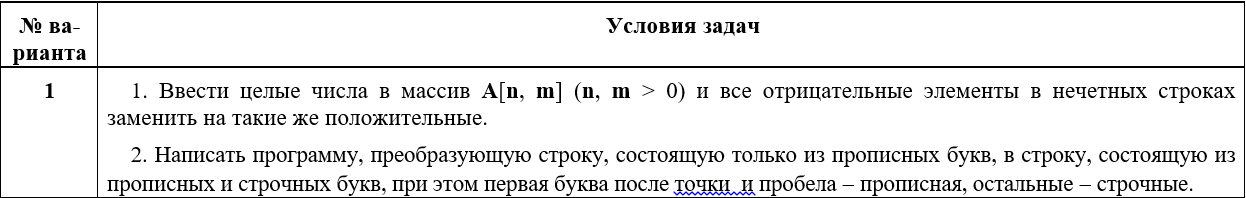
}

} while (true); // Бясконцы цыкл

}



Варыянт 1



Заданне 1, 2

#include <iostream>

#include <string> // Бібліятэка для задання з сказам

#include <Windows.h>

using namespace std;

void zamenaAdmounix(int\*\* A, int n, int m) // Функцыя для замены адмоўных элементаў у матрыцы

{

int i, j;

for (i = 0; i < n; i++) { // Замяняем элементы

for (j = 0; j < m; j++) {

if (i % 2 != 0 && A[i][j] < 0) {

A[i][j] \*= -1;

}

}

}

cout << "Новая матрыца: " << endl;

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

for (j = 0; j < m; j++) {

cout << A[i][j] << "\t";

}

cout << endl;

}

}

void zmenaRadka(char\* str, int size) { // Функцыя для змены літар у радку

int i;

for (i = 0; i < size; i++) {

if (str[i] == '.' || str[i] == ' ') { // Калі сустракаюцца кропка або прабел, то змянем літару пасля іх

str[i + 1] = toupper(str[i + 1]);

}

}

cout << "Новы радок: \n" << str << endl;

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

char exitWord[10] = "знічка"; // Калі будзе ўведзена гэтае слова, праграма скончыцца

char userInput[10];

do {

int choice = 0;

bool validChoice = false; // Ставім сцяг

do {

cout << "Выбярыце заданне (1 альбо 2): ";

cin >> choice;

if (cin.fail() || choice < 1 || choice > 2) { // Праверка на правільнасць увода

cin.clear();

cin.ignore(1000, '\n');

cout << "Памылка. Калі ласка, выбярыце нумар задання ад 1 да 2: " << endl;

}

else {

validChoice = true;

}

} while (!validChoice);

cin.clear();

switch (choice)

{

case 1: {

int n, m, i, j;

cout << "Увядзіце памер радкоў N: "; cin >> n;

cout << "Увядзіце памер слупкоў M: "; cin >> m;

int\*\* A;

A = new int\* [n];

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

A[i] = new int[m];

cout << "Цвядзіце элементы матрыцы: " << endl;

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

for (j = 0; j < m; j++) {

cin >> A[i][j];

}

}

cout << "Новая матрыца матрыца: " << endl;

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

for (j = 0; j < m; j++) {

cout << A[i][j] << "\t";

}

cout << endl;

}

zamenaAdmounix(A, n, m);

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

delete A[i];

delete[] A; // Ачышчаем памяць, якую выдзялілі для двумернага масіва

break;

}

case 2: {

int size = 0;

cout << "Увядзіце памер радка: "; cin >> size;

cin.ignore();

char\* str = new char[size + 1]; // Выдзяленне памяці для радка

cout << "Увядзіце радок: ";

cin.getline(str, size + 1); // Цалкам чытаем радок

zmenaRadka(str, size + 1);

delete[] str; // Ачышчаем памяць

}

}

cout << "Увядзіце слова \"" << exitWord << "\", калі хочаце скончыць праграму, калі не, што заўгодна іншае: ";

cin.ignore();

cin.getline(userInput, sizeof(userInput)); // Чытаем радок карыстальніка

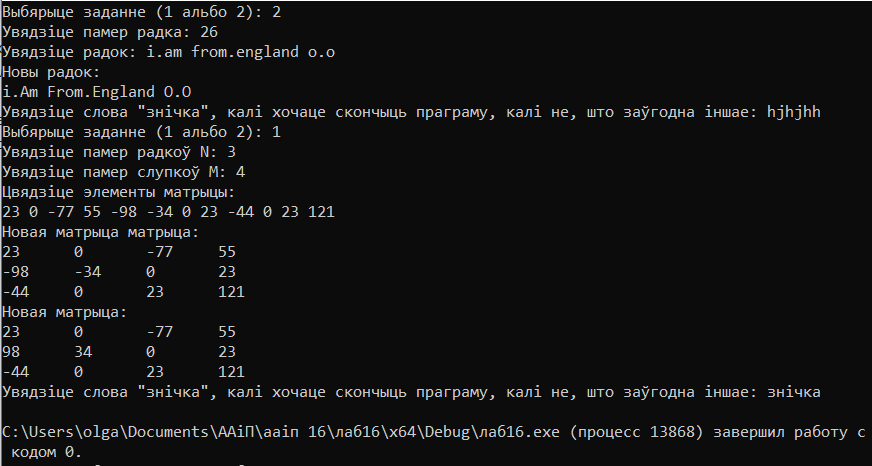
if (strcmp(userInput, exitWord) == 0) { // Параўноваем то, што было ўведзена, з тым, што патрабуецца для таго, каб скончыць праграму

break;

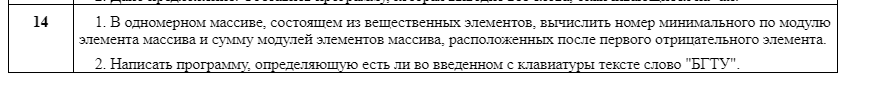
}

} while (true); // Бясконцы цыкл

}



Варыянт 14



Заданне 1, 2

#include <iostream>

#include <string> // Бібліятэка для задання з сказам

#include <Windows.h>

using namespace std;

void poshuk(char\*\* A, int wordCount, int wordSize)

{

bool hasBSTU = false;

for (int i = 0; i < wordCount; i++) // Для кожнага слова

{

for (int j = 0; j < wordSize - 3; j++)

{

if (A[i][j] == 'Б' and A[i][j + 1] == 'Д' and A[i][j + 2] == 'Т' and A[i][j + 3] == 'У') // Шукаем БДТУ

{

hasBSTU = true;

}

}

}

if (hasBSTU)

{

cout << "БДТУ ёсць у тэксце" << endl;

}

else

{

cout << "БДТУ у тэксце няма" << endl;

}

}

void forMatrizca(float\* A, int size) {

float minElement = abs(A[0]);

bool wasNegative = false;

int minIndex;

float sum = 0;

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

{

if (abs(A[i]) < minElement) // Знаходзім мінімальны па модулі элемент

{

minElement = abs(A[i]);

minIndex = i;

}

if (wasNegative) // Калі быў адмоўны лік

{

sum += A[i];

}

if (A[i] < 0)

{

wasNegative = true; // Запамінаем, калі знайшоўся адмоўны элемент

}

}

cout << "Найменшы элемент па модулі мае нумар: " << minIndex << ", сума элементаў пасля першага адмоўнага: " << sum;

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

char exitWord[10] = "анёл"; // Калі будзе ўведзена гэтае слова, праграма скончыцца

char userInput[10];

do {

int choice = 0;

bool validChoice = false; // Ставім сцяг

do {

cout << "Выбярыце заданне (1 альбо 2): ";

cin >> choice;

if (cin.fail() || choice < 1 || choice > 2) { // Праверка на правільнасць увода

cin.clear();

cin.ignore(1000, '\n');

cout << "Памылка. Калі ласка, выбярыце нумар задання ад 1 да 2: " << endl;

}

else {

validChoice = true;

}

} while (!validChoice);

cin.clear();

switch (choice)

{

case 1: {

int size;

cout << "Увядзіце размер матрыцы: ";

cin >> size;

if (size > 0)

{

float\* A = new float[size];

cout << "Увядзіце элементы масіву: " << endl ;

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

{

cin >> A[i];

}

forMatrizca(A, size);

}

break;

}

case 2: {

cout << "Увядзіце колькасць слоў і іх памер: " << endl;

int wordCount, wordSize;

cin >> wordCount >> wordSize;

char\*\* A = new char\* [wordCount]; // Робім двумерны масіў

cout << "Увядзіце словы: ";

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

{

A[i] = new char[wordSize]; // Робім масіў патрэбнага памера для захоўвання слова

cin >> A[i];

}

poshuk(A, wordCount, wordSize);

}

}

cout << "Увядзіце слова \"" << exitWord << "\", калі хочаце скончыць праграму, калі не, што заўгодна іншае: ";

cin.ignore();

cin.getline(userInput, sizeof(userInput)); // Чытаем радок карыстальніка

if (strcmp(userInput, exitWord) == 0) { // Параўноваем то, што было ўведзена, з тым, што патрабуецца для таго, каб скончыць праграму

break;

}

} while (true); // Бясконцы цыкл

}

