|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |
| Институт искусственного интеллекта |
| Кафедра технологий искусственного интеллекта |

ПРАКТИЧЕСКАЯ РАБОТА № 10

по дисциплине

«Процедурное программирование»

Обучающийся Сысоенко Глеб Максимович

Группа КАБО-01-23

Руководитель *Яковлев Д. А*

Москва 2023

**Практическая работа №10**

**Тема:** «Динамические массивы и функции»

**Цель практической работы:**

Разработка программ языке С с использованием двухмерных массивов, включая обработку главной и побочной диагонали, треугольную часть.

**Описание:**

Выделите память для массива чисел и заполнить его любым способом. Создайте

функцию, выполняющую задачу, указанную в варианте. Вызовите её из основной

функции, получите её результат и выведите его

**Задание 1:**

Вариант 3. Дан двумерный массив a, размером (n´m) и одномерный массив х(n). Четные столбцы матрицы заменить массивами х. (не использовать %, использовать for).

**Код программы 1:**

//Вариант 3. Дан двумерный массив a, размером (n´m) и

// одномерный массив х(n). Четные столбцы матрицы заменить

// массивами х. (не использовать %, использовать for).

#include <unistd.h>

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <time.h>

#include "math.h"

int\*\* beria(int \*\*A, int \*C, int \*\*B){

int i, x, y;

int \*ZN;

x = C[0];

y = C[1];

printf("\n\n");

printf("RAZMERNOST VIBR MATRICI X - > |%4d| Y - > |%4d|\n", C[0], C[1]);

printf("Process runing");

for (i = 0; i < 4; i++){

printf(".");

usleep(1250000);}

ZN = calloc(x, sizeof(int));

printf("\n\n");

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

ZN[i] = (-100 + rand() % 201);

printf("|%4d|", ZN[i]);

}

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

if ((i + 1) % 2 == 0){

for (int jj = 0; jj < y; jj++){

A[jj][i] = ZN[jj];

}

}

}

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

for (int j = 0; j < x; j++){

B[i][j] = A[i][j];

}

}

printf("\n\n");

return B;

}

int main() {

srand(time(NULL));

int \*\*A, \*C, \*\*B;

int i, hehe, x, y;

printf("VVEDITE RAZMER PO X (MAX 10)");

scanf("%d", &x);

printf("VVEDITE RAZMER PO Y (MAX 10)");

scanf("%d", &y);

// Контроль ввода

if (x > 10 || y > 10){

printf("BRAT TAK NELZA \nProgramm shutdown");

for (i = 0; i < 4; i++){

printf(".");

usleep(1250000);

}

exit(1);

}

A = calloc(y, sizeof(int\*));

C = calloc(2, sizeof(int));

C[0] = x;

C[1] = y;

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

A[i] = calloc(x, sizeof(int));

}

B = calloc(y, sizeof(int\*));

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

B[i] = calloc(x, sizeof(int));

}

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

for (int j = 0; j < x; j++) {

A[i][j] = (-100 + rand() % 201);

}

}

int nline = (x < 15) ? x : 15;

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

for (int j = 0; j < x; j++) {

printf("|%4d|%s", A[i][j], ((j + 1) % nline) ? "" : "\r\n");

}

}

beria(A, C, B);

usleep(1250000);

printf("\n");

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

for (int j = 0; j < x; j++) {

printf("|%4d|%s", B[i][j], ((j + 1) % nline) ? "" : "\r\n");

usleep(4000);

}

}

printf("Programm shutdown");

for (i = 0; i < 4; i++){

printf(".");

usleep(1250000);

}

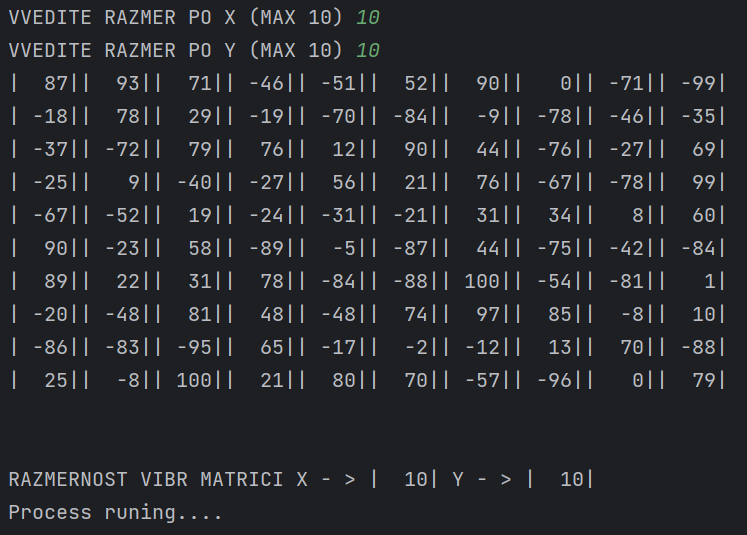
free(A);

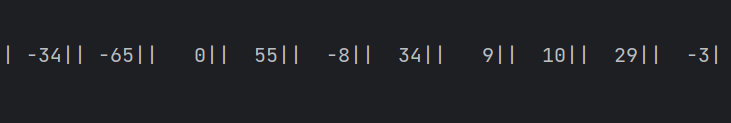
free(B);

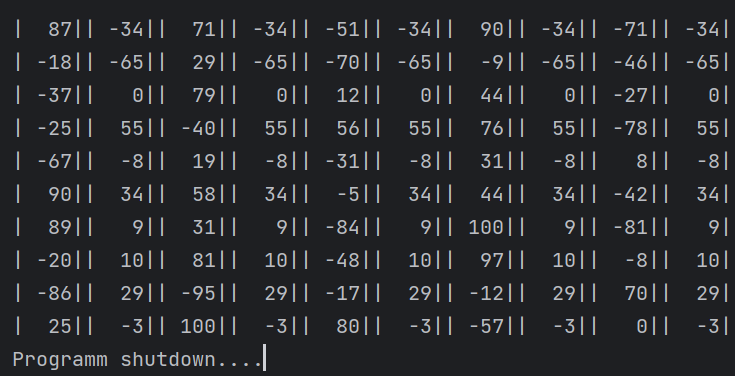
free(C);

}

**Скриншот выполнения программы 1:**







**Задание 2:**

Вариант 3. Дан двумерный массив A, размером (n´n) (или квадратная матрица A). Найти сумму номеров минимального и максимального элементов её главной диагонали.

**Код программы 2:**

//Вариант 3. Дан двумерный массив A, размером (n´n)

//(или квадратная матрица A). Найти сумму номеров минимального

//и максимального элементов её главной диагонали.

#include <unistd.h>

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <time.h>

#include "math.h"

int\*\* beria(int \*\*A, int \*C, int \*B){

int i, x, y, sr;

x = C[0];

y = C[1];

printf("\n\n");

printf("RAZMERNOST VIBR MATRICI X - > |%2d| Y - > |%2d|\n", C[0], C[1]);

printf("Process runing");

for (i = 0; i < 4; i++){

printf(".");

usleep(1250000);}

printf("\n\n");

int jj = 100;

int bj = -100;

int \*R1, \*R2;

R1 = calloc(x, sizeof(int));

R2 = calloc(x, sizeof(int));

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

printf("ZN STROK -> |%2d|\n", A[i][i]);

if (A[i][i] < jj){

jj = A[i][i];

R1[i] = i;}

if (A[i][i] > bj){

bj = A[i][i];

R2[i] = i;}

}

printf("\nMIN EL - > |%2d| MAX EL - > |%2d|\n\n", jj, bj);

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

printf("|%2d|", R1[i]);}

printf("\n\n");

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

printf("|%2d|", R2[i]);}

int xex, xex2;

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

if (A[i][i] == jj){

xex = R1[i];}

if (A[i][i] == bj){

xex2 = R2[i];}}

sr = xex + xex2;

B[0] = xex;

B[1] = xex2;

B[2] = sr;

}

int main() {

srand(time(NULL));

int \*\*A, \*C, \*B;

int i, hehe, x, y;

printf("VVEDITE RAZMER MATRICI (MAX 10)");

scanf("%d", &x); y = x;

// Контроль ввода

if (x > 10 || y > 10){

printf("BRAT TAK NELZA \nProgramm shutdown");

for (i = 0; i < 4; i++){

printf(".");

usleep(1250000);

} exit(1); }

A = calloc(y, sizeof(int\*));

C = calloc(2, sizeof(int));

C[0] = x;

C[1] = y;

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

A[i] = calloc(x, sizeof(int));}

B = calloc(3, sizeof(int));

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

for (int j = 0; j < x; j++) {

A[i][j] = (-100 + rand() % 201);}}

printf("\n\n");

int nline = (x < 15) ? x : 15;

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

for (int j = 0; j < x; j++) {

printf("|%4d|%s", A[i][j], ((j + 1) % nline) ? "" : "\r\n");}}

beria(A, C, B);

usleep(1250000);

printf("\n\n");

printf("ADR MIN EL - > |%2d| ADR MAX EL - > |%2d| IX SUMA - > |%2d|", B[0], B[1], B[2]);

printf("\n\nProgramm shutdown");

for (i = 0; i < 4; i++){

printf(".");

usleep(1250000);}

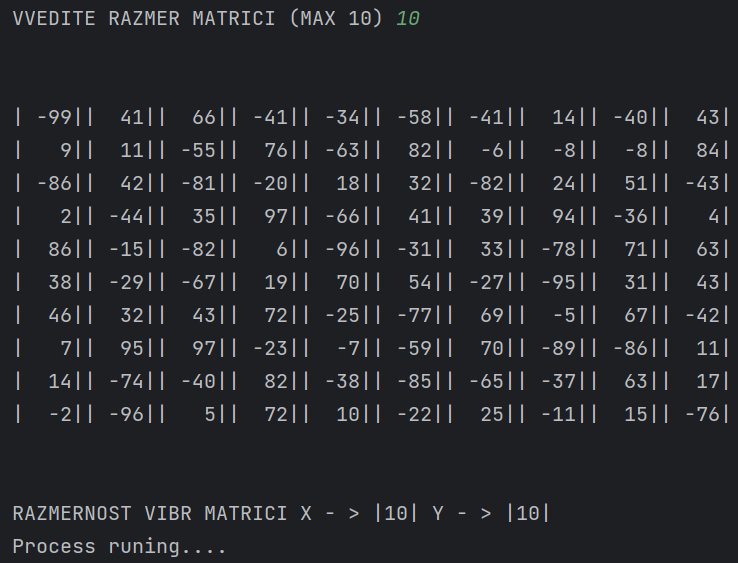
free(A);

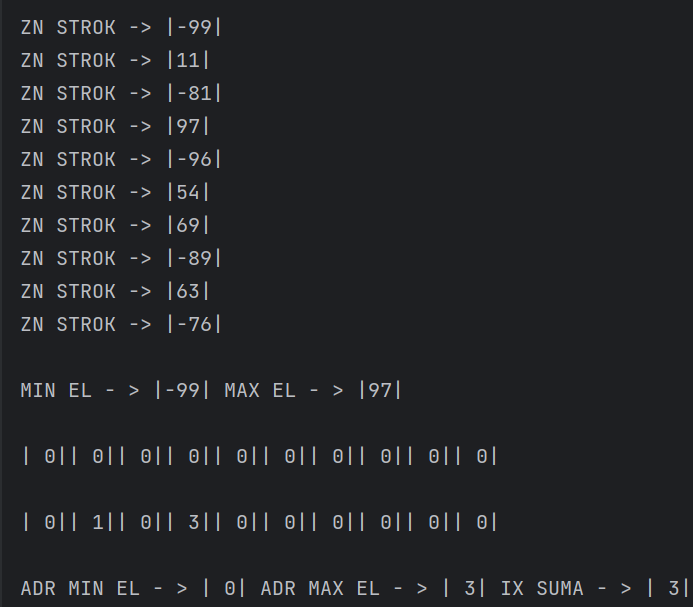
free(B);

free(C);

}

**Скриншот выполнения программы 2:**





**Задание 3:**

Вариант 3. Дан двумерный массив А, размером (n´n) (или квадратная матрица А). Найти среднее арифметическое положительных элементов, нижней треугольной матрицы, расположенной ниже главной диагонали, исключая саму главную диагональ.

**Код программы 3:**

//Вариант 3. Дан двумерный массив А, размером (n´n) (или квадратная матрица А).

// Найти среднее арифметическое положительных элементов, нижней треугольной матрицы,

// расположенной ниже главной диагонали, исключая саму главную диагональ.

#include <unistd.h>

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <time.h>

#include "math.h"

int\*\* beria(int \*\*A, int \*C, int \*B){

int i, x, y, sum = 0, ch = 0, sr;

int \*ZN;

x = C[0];

y = C[1];

printf("\n\n");

printf("RAZMERNOST VIBR MATRICI X - > |%2d| Y - > |%2d|\n", C[0], C[1]);

printf("Process runing");

for (i = 0; i < 4; i++){

printf(".");

usleep(1250000);}

printf("\n\n");

int jj = 100;

int bj = -100;

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

for (int j = 0; j < y; j ++){

if (i != j && i < j){

printf("|%3d|", A[j][i]);

sum += A[j][i];

ch += 1;

}

}

}

// printf("\n%d %d", bj, jj); Разменование это получение значения по адресу, который содержит указатель

sr = sum / ch;

B[0] = sr;

B[1] = sum;

B[2] = ch;

}

int main() {

srand(time(NULL));

int \*\*A, \*C, \*B;

int i, hehe, x, y;

printf("VVEDITE RAZMER PO X (MAX 10)");

scanf("%d", &x);

printf("VVEDITE RAZMER PO Y (MAX 10)");

scanf("%d", &y);

// Контроль ввода

if (x > 10 || y > 10){

printf("BRAT TAK NELZA \nProgramm shutdown");

for (i = 0; i < 4; i++){

printf(".");

usleep(1250000);

}

exit(1);

}

if (x != y){

printf("\nREZULT BUDDET NE KORETEN");

}

A = calloc(y, sizeof(int\*));

C = calloc(2, sizeof(int));

C[0] = x;

C[1] = y;

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

A[i] = calloc(x, sizeof(int));

}

B = calloc(3, sizeof(int));

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

for (int j = 0; j < x; j++) {

A[i][j] = (-100 + rand() % 201);

}

}

printf("\n\n");

int nline = (x < 15) ? x : 15;

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

for (int j = 0; j < x; j++) {

printf("|%4d|%s", A[i][j], ((j + 1) % nline) ? "" : "\r\n");

}

}

beria(A, C, B);

usleep(1250000);

printf("\n");

printf("SRED ARF - > |%2d| SUMA - > |%2d| KOL-VO - > |%2d|", B[0], B[1], B[2]);

printf("\n\nProgramm shutdown");

for (i = 0; i < 4; i++){

printf(".");

usleep(1250000);

}

free(A);

free(B);

free(C);

}

**Скриншот выполнения программы 3:**

A screenshot of a computer screen

Description automatically generated

A black background with many small colored lines

Description automatically generated with medium confidence

**Задание 4:**

Вариант 3. Дано два двумерных массива х(n1´m) ,y(n2´m). Cформировать массив z((n1+n2+2)´m), таким образом, чтобы сначала располагались n1 строк из массива x, затем две строки заполненные нулями, далее n2 строк из массива y.

**Код программы 4:**

//Вариант 3. Дано два двумерных массива х(n1´m) ,y(n2´m).

// Cформировать массив z((n1+n2+2)´m), таким образом, чтобы сначала располагались

//n1 строк из массива x, затем две строки заполненные нулями, далее n2 строк из массива y.

#include <unistd.h>

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <time.h>

#include "math.h"

int\*\* beria(int \*\*A, int \*C, int \*\*B){

int i, x, y, y2, y3 = 2, j;

int \*\*ZN, \*\*ZN2;

x = C[0];

y = C[1];

y2 = C[2];

int nline = (x < 15) ? x : 15;

printf("\n");

printf("RAZMERNOST VIBR MATRICI X - > |%2d| Y1 - > |%2d| Y2 - > |%2d|\n", C[0], C[1], C[2]);

printf("Process runing");

for (i = 0; i < 4; i++){

printf(".");

usleep(1250000);}

ZN = calloc(y3, sizeof(int\*));

ZN2 = calloc(y + y2 + y3, sizeof(int\*));

//СОЗДАНИЕ НУЛЕВОГО МАССИВА

for (i = 0; i < y3; i++) {

ZN[i] = calloc(x, sizeof(int));}

//СОЗДАНИЕ МАССИВА ЗАПИСИ

for (i = 0; i < y + y2 + y3; i++) {

ZN2[i] = calloc(x, sizeof(int));}

//ЗАПИСЬ В ОБЩИЙ МАССИВ МАССИВА А

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

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

ZN2[i][j] = A[i][j];}}

//ЗАПИСЬ В ОБЩИЙ МАССИВ МАССИВА ЗН(НУЛЕВОГО)

for (i = 0; i < y3; i++) {

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

ZN2[y + i][j] = ZN[i][j];}}

//ЗАПИСЬ В ОБЩИЙ МАССИВ МАССИВА В

for (i = 0; i < y2; i++) {

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

ZN2[y + y3 + i][j] = B[i][j];}}

printf("\n\nMAS OBZHI\n\n");

for (i = 0; i < y3+y+y2; i++) {

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

printf("|%4d|%s", ZN2[i][j], ((j + 1) % nline) ? "" : "\r\n");}}

}

int main() {

srand(time(NULL));

int \*\*A, \*C, \*\*B;

int i, hehe, x, y, y2;

printf("VVEDITE RAZMER PO X (MAX 10)");

scanf("%d", &x);

printf("VVEDITE RAZMER PO Y1 (MAX 10)");

scanf("%d", &y);

printf("VVEDITE RAZMER PO Y2 (MAX 10)");

scanf("%d", &y2);

// Контроль ввода

if (x > 10 || y > 10 || y2 > 10){

printf("BRAT TAK NELZA \nProgramm shutdown");

for (i = 0; i < 4; i++){

printf(".");

usleep(1250000);}

exit(1);}

A = calloc(y, sizeof(int\*));

C = calloc(3, sizeof(int));

C[0] = x;

C[1] = y;

C[2] = y2;

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

A[i] = calloc(x, sizeof(int));}

B = calloc(y2, sizeof(int\*));

for (i = 0; i < y2; i++) {

B[i] = calloc(x, sizeof(int));}

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

for (int j = 0; j < x; j++) {

A[i][j] = (-100 + rand() % 201);}}

for (i = 0; i < y2; i++) {

for (int j = 0; j < x; j++) {

B[i][j] = (-100 + rand() % 201);}}

printf("\nMAS A\n\n");

int nline = (x < 15) ? x : 15;

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

for (int j = 0; j < x; j++) {

printf("|%4d|%s", A[i][j], ((j + 1) % nline) ? "" : "\r\n");}}

printf("\nMAS B\n\n");

for (i = 0; i < y2; i++) {

for (int j = 0; j < x; j++) {

printf("|%4d|%s", B[i][j], ((j + 1) % nline) ? "" : "\r\n");}}

beria(A, C, B);

printf("\n\nProgramm shutdown");

for (i = 0; i < 4; i++){

printf(".");

usleep(1250000);}

free(A);

free(B);

free(C);

}

**Скриншот выполнения программы 4:**

A screenshot of a computer

Description automatically generated

A screenshot of a computer code

Description automatically generated