МИНИСТЕРСТВО ТРАНСПОРТА И КОММУНИКАЦИЙ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования

«Белорусский государственный университет транспорта»

Кафедра «Информационно-управляющие системы и технологии»

Отчет  
по практическим работам  
по дисциплине «Основы алгоритмизации и программирования»

Выполнил Проверил

студент группы ГИ-11 ст. преп. каф. ИУСиТ

Зайцев И. С. Голдобина Т. А.

Гомель, 2019

Содержание

[Практическая работа №12 РАЗРАБОТКА АЛГОРИТМА, СОСТАВЛЕНИЕ, ОТЛАДКА И ВЫПОЛНЕНИЕ ПРОГРАММЫ ОБРАБОТКИ ДВУМЕРНОГО МАССИВА 3](#_Toc28092699)

[Задания 3](#_Toc28092700)

[Контрольные вопросы 9](#_Toc28092701)

# Практическая работа №13 Применение указателей для обработки массивов

### Цель

Применить указатели для обработки массивов.

## Задания

### Задание 1

### Условие

Переработать задание 1 из ПР11 с применением указателей (в т.ч. индексированных указателей).

### Программный код

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main()

{ int \*q,\*w,\*e,\*r,\*t,\*y,\*u;

int U[10], min=0;

int i, D[10], n,x=1,z;

printf ("Enter n:");

scanf ("%d", &n);

q=U;

w=&min;

e=&i;

r=D;

t=&n;

y=&x;

u=&z;

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

{

printf("U[%d] = ", \*e);

scanf("%d", &U[i]);

}

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

{D[i]=(1/tan(U[i]))/(n+1)-log(abs(n\*U[i]));}

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

{

printf("D[%d]=%d\n",\*e, \*r);

\*r++;

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

if (D[i]<0)

{

\*y\*=D[i];

}

} printf ("%d\n",\*y);

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

if (\*w>=D[i]) {\*w=D[i];}

}printf ("%d\n",\*w);

\*u=D[9];

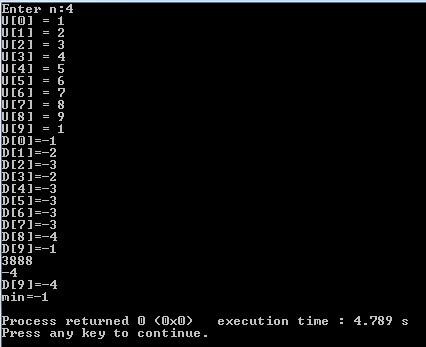
D[9]=\*w;

\*w=\*u;

printf ("D[9]=%d\n", \*(D + 9));

printf ("min=%d\n", \*w);

return 0;}

  
Рисунок 1 − Решение задания 1

### Задание 2

### Условие

### Переработать задание 2 из ПР12 с применением указателей (в т.ч. индексированных указателей).

### Программный код

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include <locale.h>

#include <string.h>

int main()

{setlocale(LC\_ALL,"");

int \*q,\*w,\*e,\*r,\*t;

float \*l,\*v, \*mass;

int n,i=0,j,k=2,min1;

float o=0,max;

printf("n=");

scanf("%d",&n);

float P[n][n];

q=&n;

w=&i;

e=&j;

r=&k;

t=&min1;

l=&o;

v=&max;

mass=P;

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

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

if (\*e<0)\*mass=pow(tan(1),2)-sqrt(1);

else \*mass=pow(tan(i),2)-sqrt(j);

printf("%.2f\t",\*mass);

\*mass++;

}printf("\n");}

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

\*l+=P[i][1];}

printf("Cумма элементов второго столбца: %.2f\n",\*l);

\*v=P[1][0];

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

if(P[1][j]>\*v)\*v=P[1][j];}

printf("Максимальный элемент второй строки:%.2f\n",\*v);

\*t=P[0][0];

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

for(j=0;j<10;j++){

if(P[i][j]<\*t){\*t=P[i][j];}}}

printf("%lf\n",\*t);

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

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

if(P[i][j]<0){P[i][j]=2;}

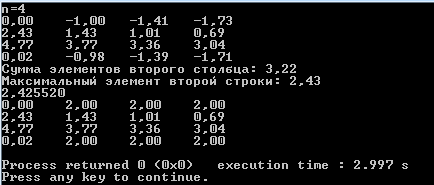
printf("%.2f\t",P[i][j]);

}

printf("\n");

}

return 0;}

  
Рисунок 2 − Решение задания 2

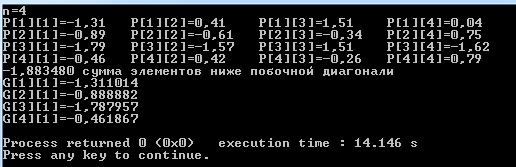
### Задание 3

### Условие

### Переработать задание 3 из ПР12 с применением указателей (в т.ч. индексированных указателей).

### Программный код

#include <stdio.h>  
#include <stdlib.h>  
#include <math.h>  
#include <locale.h>  
#include <time.h>  
  
int main()  
{setlocale(LC\_ALL,"");  
int n, \*q, i, \*w, j, \*e, k;  
q=&n;  
printf("n=");  
scanf("%d", q);  
float min=0,sum=0, men=-2, max=2, P[n][n];  
float \*z, \*t, \*x, \*c, \*mass;  
w=&i;  
e=&j;  
z=&min;  
t=&sum;  
x=&men;  
c=&max;  
mass=P;  
  
srand(time(NULL));  
for(\*w=0; \*w<\*q; (\*w)++){  
for(\*e=0; \*e<\*q; (\*e)++){  
P[\*w][\*e]=(float)rand()/RAND\_MAX\*(\*c-\*x)+\*x;  
printf("P[%d][%d]=%.2f\t", \*w+1, \*e+1, P[\*w][\*e]);  
if (\*w+\*e+1>\*q && P[\*w][\*e]<0){\*t+=P[\*w][\*e];}  
if(\*w==\*e && P[\*w][\*e]<=min){\*z=P[\*w][\*e];  
k=j;}}  
printf("\n");  
}  
if (\*t==0) {printf("нет отрицательных ниже побочной диагонали\n");}  
else {printf("%f сумма элементов ниже побочной диагонали\n", \*t);}  
  
for(\*w=0; \*w<\*q; (\*w)++){  
printf("G[%d][%d]=%f\n",\*w+1,k+1,P[\*w][k]);  
}  
return 0;  
}

  
Рисунок 3 − Решение задания 3

## Контрольные вопросы

1. Понятие указателя.

**Указатель** — переменная, содержащая адрес объекта. Указатель не несет информации о содержимом объекта, а содержит сведения о том, где размещен объект.

1. Как объявить указатель на вещественную переменную двойной точности?

double a, \*b;

b=&a;

1. Операторы для работы с указателями.

Имеется два специальных оператора для работы с указателями - \* и &. Оператор & - это унарный оператор, возвращающий адрес операнда. Оператор \* - это унарный оператор, возвращающий значение переменной, находящейся по указанному адресу.

1. Понятие и назначение пустого указателя.

Константа нулевого указателя — это целочисленное константное выражение со значением 0 или (только в Си) такое же выражение, но приведённое к типу void\*.  Константа нулевого указателя, приведённая к любому типу указателей, является **нулевым указателем**.

Нулевые указатели придуманы как удобный способ «отметить» указатели, которые заведомо не указывают на корректный адрес в памяти. Например, при объявлении указателя как автоматической переменной его значение не определено. Чтобы отметить, что этот указатель ещё не содержит корректный адрес в памяти, такому указателю присваивают константу нулевого указателя.

1. Вывод значения указателя на экран

int a;   //переменная  
char \*b; //указатель  
b = &a;  // p = адрес a

printf(" Данные по адресу указателя b равны %d", \*b);

1. Как обратиться к элементам массива с помощью указателей?

int i,j; // индексные переменные

int N, m[3][6] = { { 1, 2, 3, 4, 5, 6 },

{ 7, 8, 9, 10, 11, 12 },

{ 13, 14, 15, 16, 17, 18 }

};

i = 2; j = 4;

N = m[i][j];     /\* N получит значение элемента, стоящего на пересечении 2-ой строки и 4-го столбца\*/

N = m[i-2][2\*j-3]; /\* N получит значение элемента нулевой строки и 5-го столбца\*/

1. Индексирование указателей.

Int \*р, i [10];  
p = i;  
р[5] = 100; // присвоение с помощью индекса   
\*(р+5) = 100; /\* присвоение с помощью арифметики с указателями \*/

### Выводы по работе

Применил указатели для обработки массивов.