МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра системного програмування і спеціалізованих комп‘ютерних систем

**Лабораторна робота №4**

**з дисципліни: «Програмування»**

**тема: «Блочна структура програми. Процедури та функції»**

**Варіант №22**

Виконав:

Студент групи КВ-51

Базильський Л.О.

Перевірив:

Київ

2015

**Постановка задачі**

Написати програму рішення завдання з обов’язковим використанням процедур та/або функцій для опису дій, які повинні повторюватися, а саме:

* введення елементів масиву;
* виведення елементів масиву;
* використання над масивом дій, заданих за варіантом.

**Вимоги до програми**

1. Алгоритми усіх процедур/функцій не повинні залежати від глобальних даних, тобто масиви та інші дані, які передаються до процедури/функції та повертаються з них, повинні передаватися як параметри або значення (результат) функції.
2. Використання видів параметрів у процедурах/функціях повинно бути логічним з точки зору швидкодії програми та економності використання оперативної пам’яті комп’ютера.
3. В тестах значення елементів масивів підібрати таким чином, щоб в алгоритмі обробки масиву були перевірені всі ситуації, які можуть виконувати для заданого алгоритма, тобто продемонструвати повну коректність та універсальність алгоритма.

**Варіант №22:** Дано три матриці цілих чисел A[4,4], B[9,5], C[6,6]. При обході по стовпцях в кожній матриці знайти перший максимальний і останній мінімальний елемент, а також поміняти їх місцями.

Program Lab4;

Const iLenA=4;jLenA=4;

iLenB=9;jLenB=5;

iLenC=6;jLenC=6;

Type DynMatrix=array of array of integer;

Var A,B,C:DynMatrix;

Procedure MATRIX\_VVOD(var name:DynMatrix; iLen:integer; jLen:integer);

var i,j:byte;

begin

randomize;

SetLength(name, iLen);

for i:=0 to iLen-1 do

begin

SetLength(name[i], jLen);

for j:=0 to jLen-1 do name[i,j]:=random(101)-50;

end;

end;

Procedure MATRIX\_MINMAX\_TRANS(var name:DynMatrix; iLen:integer; jLen:integer);

var i,j:byte;

xMax,yMax,xMin,yMin:byte;

begin

for j:=0 to jLen-1 do

begin

xMax:=0;yMax:=j;

xMin:=0;yMin:=j;

for i:=1 to iLen-1 do

begin

if name[i,j]>name[xMax,yMax] then xMax:=i

else

if name[i,j]<=name[xMin,yMin] then xMin:=i

end;

name[xMax,yMax]:=name[xMax,yMax]+name[xMin,yMin];

name[xMin,yMin]:=name[xMax,yMax]-name[xMin,yMin];

name[xMax,yMax]:=name[xMax,yMax]-name[xMin,yMin];

end;

end;

Procedure MATRIX\_DISPLAY(name:DynMatrix; iLen:integer; jLen:integer);

var i,j:byte;

begin

for i:=0 to iLen-1 do

begin

for j:=0 to jLen-1 do write(name[i,j], ' ');

writeln;

end;

end;

Begin

writeln('Початкова матриця A:');

MATRIX\_VVOD(A,iLenA,jLenA);

MATRIX\_DISPLAY(A,iLenA,jLenA);

writeln;

writeln('Перетворена матриця A:');

MATRIX\_MINMAX\_TRANS(A,iLenA,jLenA);

MATRIX\_DISPLAY(A,iLenA,jLenA);

writeln;

writeln('Початкова матриця B:');

MATRIX\_VVOD(B,iLenB,jLenB);

MATRIX\_DISPLAY(B,iLenB,jLenB);

writeln;

writeln('Перетворена матриця B:');

MATRIX\_MINMAX\_TRANS(B,iLenB,jLenB);

MATRIX\_DISPLAY(B,iLenB,jLenB);

writeln;

writeln('Початкова матриця C:');

MATRIX\_VVOD(C,iLenC,jLenC);

MATRIX\_DISPLAY(C,iLenC,jLenC);

writeln;

writeln('Перетворена матриця C:');

MATRIX\_MINMAX\_TRANS(C,iLenC,jLenC);

MATRIX\_DISPLAY(C,iLenC,jLenC);

End.





