Міністерство освіти і науки, молоді та спорту України

Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

**Лабораторна робота №8**

З дисципліні «Програмування»

**Тема: «Робота з файлами»**

Виконав: Перевірив:

Студент групи ІО-33 Викладач

Заколенко Р. К. Саверченко В. Г.

Київ 2013

**Завдання:**

1. Отримати елементи квадратної матриці ***А=⎜ a i,j ⎜,******i,j=1, ..., n, a i,j = f(i, j),*** *де*

***f(i, j) = -2\*( j / 2 - 3.1 )\*( i - 3.9 ) + 2.5 i \* j –10.***

1. Вивести компоненти отриманої матриці.
2. Елементи квадратної матриці записати у файл.
3. Дані з файла записати у матрицю ***В***.
4. Із матриці ***В*** отримати компоненти вектора ***X={xi}, i=1,2,...,n*** за наступним правилом: у ролі вектора прийняти головну діагональ матриці , яка перетворена наступним чином: на початку кожного рядка повинні бути її невід’ємні елементи, а у кінці - від’ємні (із збереженням порядку слідування тих чи інших елементів).
5. Вивести компоненти отриманого вектора.
6. Перетворити вектор за наступним правлом: **i-й** елемент вектора дорівнює ***(хi - хi+1 )/2,*** де *.*
7. Вивести компоненти перетвореного вектора.
8. Об'єднати рішення попередніх завдань в одну програму, оформивши ці задачі з використанням функцій і процедур c параметрами.
9. Процедури і функції повинні входити до складу модуля. Основна програма повинна складатися тільки з операторів процедур і функцій.

**Код модуля:**

Unit Lab8;

Interface

Type

Tindex=0..100;

Tarray=array[Tindex] of real;

Matrix=array[Tindex] of Tarray;

function f(i,j:Tindex):real;

Procedure InputMatrix(var n:Tindex; var A:Matrix);

Procedure OutputMatrix(n:Tindex; A:Matrix);

Procedure WriteMatrix(var mf:text; A:Matrix;n:Tindex);

Procedure ResetMatrix(mf:text; var B:Matrix;n:Tindex);

Procedure Vektor(B:Matrix;var X:Tarray;n:Tindex);

Procedure OutputVektor(X:Tarray;n:Tindex);

Procedure NewVektor(var X:Tarray; var X1:Tarray; n:Tindex);

Implementation

var

i,j:Tindex;

Function f(i,j:Tindex):real;

begin

f:=-2\*(j/2-3.1)\*(i-3.9)+exp(ln(2.5)\*i)\*j-10;

end;

Procedure InputMatrix(var n:Tindex; var A:Matrix);

begin

write('enter the size of matrix, n=');

read(n);

for i:=1 to n do

for j:=1 to n do

A[i][j]:=f(i,j);

end;

Procedure OutputMatrix(n:Tindex; A:Matrix);

begin

writeln('Matrix A:');

for i:=1 to n do

begin

for j:=1 to n do

write(A[i][j],' ');

writeln

end;

end;

Procedure WriteMatrix(var mf:text; A:Matrix;n:Tindex);

begin

assign(mf,'D:\lab8.txt');

rewrite(mf);

for i:=1 to n do

begin

for j:=1 to n do

write(mf,A[i][j],' ');

writeln

end;

close(mf)

end;

Procedure ResetMatrix(mf:text; var B:Matrix;n:Tindex);

begin

assign(mf,'D:\lab8.txt');

reset(mf);

for i:=1 to n do

for j:=1 to n do

read(mf,B[i][j]);

close(mf)

end;

Procedure Vektor(B:Matrix;var X:Tarray;n:Tindex);

begin

j:=1;

for i:=1 to n do

if B[i][i]>=0 then

begin

X[j]:=B[i][i];

j:=j+1

end;

for i:=1 to n do

if B[i][i]<0 then

begin

X[j]:=B[i][i];

j:=j+1

end;

end;

Procedure OutputVektor(X:Tarray;n:Tindex);

begin

writeln('Vektor X:');

for i:=1 to n do

write(X[i],' ');

writeln

end;

Procedure NewVektor(var X:Tarray;var X1:Tarray; n:Tindex);

begin

for i:=1 to (n-1) do

X1[i]:=(X[i]-X[i+1])/2;

for i:=1 to n-1 do

X[i]:=X1[i];

end;

end.

**Код програми:**

Program MyMatrix;

Uses Lab8;

var

n:Tindex;

A,B:Matrix;

X,X1:Tarray;

Myfile:text;

begin

InputMatrix(n,A);

OutputMatrix(n,A);

WriteMatrix(MyFile,A,n);

ResetMatrix(MyFile,B,n);

Vektor(B,X,n);

OutputVektor(X,n);

NewVektor(X,X1,n);

OutputVektor(X,n);

end.

**Висновоки:**

1. Для отримання вектора **Х** з матриці **В** було використано два цикли **for**, перший відсортовує невід’ємні елементи головної діагоналі матриці В, другий цикл – від’ємні елементи.
2. Для перетворення вектору **Х** був використаний додатковий вектор **Х1**.