**КУРСОВА РОБОТА**

**НА ТЕМУ:**

**«Структуровані типи даних.Операції над двомірними масивами »**

**Анотація**

**В цій курсовій роботі розглянуті дії над**

**матрицями , такі як додавання , віднімання, мно-**

**ження та ділення двох матриць. А також знаход-**

**ження транспонованої та оберненої матриць.Про-**

**грами реалізовані на мові програмування Turbo**

**Pascal 7.0**

 **ЗМІСТ**

 **1.Вступ.**

 **2.Теоретична частина.**

 **2.1.Матриця і її властивості.**

 **2.2. Дії над матрицями.**

 **3.Постановка задачі.**

 **4.Додатки.**

 **4.1.Додатток 1(текст програм).**

 **4.2.Додаток 2(блок-схеми до програм).**

 **5.Висновки.**

 **6.Використана література.**

**1.Вступ.**

У всі часи людина прагнула розширити свої можливості, за допомогою різних знарядь праці, пізнання світу та засобів існування.

Так, наприклад нестачу зору компексує : мікроскоп, телескоп, радіолокатор. Обмежені можливості передачі інформації поширюються телефоном, радіо, телебаченням.

 Обчислювані машини «доповнюють» можливості мозку людини, розширюють його можливості по обробці інформації, дозволяють прискорити прийняття рішення в процесі якої-небудь роботи.

В кінці 40-х років 20 ст. Праця в області ядерної фізики, баллистики керуючих знарядь, термодинаміки і т.д. вимагали такої обчислюваної роботи, яку вже було не можливо виконати за допомогою арифмометрів-головного обчислюваного інструмента того часу. Наука і техніка були поставлені перед делемою : або всім взятись за арифмометри або винайти новий ефективний інструмент обчислення. Аналогічні проблеми уже не раз виникали, і будуть неодноразово виникати перед вченими і інженерами: екстенсивний шлях розвитку дальше неможливий, потрібний новий, інтенсивний шлях. Проблема була вирішена створенням універсальної обчислюваної машини. Термін «універсальна»використовується не випадково. Спеціалізовані машини (наприклад, для обробки банківських рахунків і т. д.) існували і раніше, але не було машини, команди якої записані в память, можна б було швидко замінити новими.

 Крім математичних обчислень ЕОМ може виконувати і логічні, тобто робити вибір між варіантами (вітками) продовження дій в залежності від виконання деяких умов. Таким чином ЕОМ-це дещо більше ніж «швидкий арифмометр».

 **Коротка характеристика різних поколінь ЕОМ**

**Перше покоління ЕОМ:**

 Технічна основа елементної бази машин 1-го покоління-електронні лампи. Максимальна швидкодія -10 у степені 2. Математичні операції в секунду(оп/с), обєм оперативної памяті -10 у 2 степені слів. Режим використання-монопольний, тобто в розпорядженні користувача були всі ресурси машини і її управління.

**Друге покоління ЕОМ:**

 Технічна основа - транзистори. максимальна швидкодія-10 у 4 степені оп/с, обєм оперативної памяті-10 у 4 степені слів.Режим виконання-пакетна обробка.

**Третє покоління ЕОМ:**

 Технічна основа-занадто великі інтегральні схеми, які на малих півпровідникових кристалах реалізують великі схеми машин 2-го покоління. Максимальна швидкодія-10 у 6 степені оп/с, оперативна память -10 у 6 степені слів, внутрішня память-10 у 9 степені слів. Метод виконання - режим розподілу часу разом з пакетною обробкою.

**4-те покоління ЕОМ:**

Технічна основа-занадто великі інтегральні схеми. Традиційна архітектура ЕОМ Фон Неймана домінувала на протязі трьох поколінь.

Максимальна швидкодія-10 у 9 степені оп/с, оперативна память-10 у 7 степені слів ,внутрішня память обмежена в основному економічними міркуваннями.

**5-те покоління ЕОМ.**

 Проекти ЕОМ п’ятого покоління знаходяться в стадії реалізації. Максимальна швидкодія математичних обчислень доповнюється тут високими скоростями логічного виводу. Форма спілкування з ЕОМ

на звичайній мові і дисципліна програм, як наука для користувача перестають в майбутньому бути актуальними.

**Історія і зміст предмета.**

 Обчислюваною математикою називають розділ математики, в якому вивчають різні проблеми одержання числових результатів обчислень математичних задач.

 Якщо звернутися до історії математики то можна помітити, що обчислювана математика перетворилась на самостійну вітку порівняно недавно, десь в середині нашого століття. Цей факт в любому напрямку науки повязані з появленням власних і внутрішніх задач.

 Обчислювальна математика, як частина математики має таку ж древню і багату історію, як і сама математика. Евклідова математика і механіка Ньютона, теорія електромагнітного поля і квантова теорія побудованіна математичній основі і дають потужні інструменти обчислень.

Зпоявленням ЕОМ розпочався золотий вік обчислювальної математики.вона швидко розвивається. Звернувшись до періоду розвитку обчислювальної математики після полявлення ЕОМ, можна побачити, що найбільш яскраві досягнення в розвязку задач були отримані саме тими вченими і інженерами, хто працював на ЕОМ, всі отрамані засоби математики:»чистої», прикладної, обчислювальної.

 З точки зору техніки обчислювальної математика дає в її розпорядження методи , які умовно можна розбити на слідуючі 4 групи: якісні, аналітичні , численні.

**2.1. Матриця і її властивості.**

 Прямокутна таблиця з m×n чисел ,що має m рядків і n стовпців

 a11 a12 ... a1n

A=... ... ... ...

 am1 am2 ... amn

називається ***матрицею***. Коротко матрицю позначають так:

 А= ai j (і=1,2,...,m; j=1,2,...,n),

де ai j - елементи матриці.

 Матрицю з єдиним стовпцем прийнято називати ***вектор-стовпцем***, а матрицю з єдиним рядком ⎯ ***вектор-рядком.***

Рівні матриці повинні мати рівні кількості рядків і стовпців, а також рівні відповідні елементи.

 Якщо в матриці число рядків рівне числу стовпців ,то матриця називається ***квадратною* :**

 a11 a12 ... a1n

A= ... ... ... ...

 an1 an2  ... ann

Матриця А\* називається ***транспонованою*** до матриці А , якщо стовпці матриці А являються рядками матриці А\*.

 Наприклад:  a11 a12

 A= a21 a22

 a31 a32

Транспонованою матрицею А\* буде:

 a11 a21 a31

A\*=

a12 a22 a32

 Приклад. Нехай А=(aij), де і=1,..,m, о=1,..,n. Це значить, що А- матриця порядку m×n. Позначимо А\* матрицю В = (bij), для якої bij = aji, тоді А\*матриця порядку n×m.

 Квадратна матриця А називається ***симетричною*** ***відносно головної діагоналі*** ,якщо ai j=aj i .

 Квадратна матриця, в якій всі елементи, що не лежать на головній діагоналі, дорівнюють нулю, називається ***діагональною.*** Якщо елементи діагональної матриці, що розміщені на головній діагоналі, дорівнюють одиниці, то матриця називається ***одиничною*** і позначають її буквою Е:

 1 0 ... 0

 Е= 0 1 ... 0

 . . . . . . . . . . .

 0 0 ... 1

**2.2. Дії над матрицями:**

 Як виявляється, над матрицями можливі арифметичні дії, властивості яких близькі до властивостей арифметичних дій над числами.

***Сумою*** двох матриць ai j і bi j з одинаковою кількістю рядків і стовпців називається матриця сi j ,у якої елементом сi j є сума aij+bij  відповідних елементів матриць ai j bi j ,тобто

 ai j + bi j  = ci j  ,

якщо ai j+bi j=ci j  (i=1,2,..,m; j=1,2,..,n)

 Приклад: a11 a12 b11 b12 a11+b11 a12+b12

 a21 a22 b21 b22 a21+b21 a22+b22

 Аналогічно знаходимо ***різницю*** двох матриць.

 Матрицці різних порядків додавати(віднімати) не можна.

 ***Множення матриці на число***. Щоб помножити матрицю на число λ або число на матрицю, потрібно кожний елемент матриці помножити на це число.

λ \* ai j  = λ ai j

 a11 a12 λ a11 λ a12

1. a21 a22 = λ a21 λ a22 .

 Безпосередніми наслідками вказаних визначень є співвідношення:

1. 1 А = А 1 = А ;
2. 0 А = А 0 = 0 ;
3. α О = О α = О ;
4. α (β А) = (α β) А = (А α) β = А (α β);
5. А + (В +С) = (А+ В) + С;
6. А + В = В + А;
7. (α + β) А = α А + β А;
8. α (А + В) = α А + α В;
9. А + О = О + А = А;
10. А + (-1)А = О;

 Тут А, В, С - матриці одного порядку, α, β - числа, О - нульова матриця (всі її елементи дорівнюють нулеві). Перевірка вказаних властивостей не викликає ускладнень.

Елемент ci j матриці С, яка є добутком матриці В на матрицю А, дорівнює сумі добутків елементів і-того рядка матриці В на відповідний елемент j-того стовпця матриці А, тобто

 k

 ci j =∑bi λaλ j (i=1,2,..,m; j=1,2,..,n).

 λ=1

 Властивості добутку матриць:

1. (А В) С = А (В С);
2. А (В + С) = А В + А С;
3. (А + В) С = А С + В С;
4. А Е = Е А = А;
5. (А В)\*= В\*А\*;

 Тут А, В, С - довільні матриці, для яких вказані рівності мають сенс.

 Доведемо першу рівність - асоціативність множення матриць.

 Позначимо D = A B, F = B C, G = D C, H = A F. Потрібно довести, що G =H. Оскільки множення вказаних вище матриць можливе, то А буде порядку m×n, В - порядку n×k, С - порядку k×l. З означення множення дістанемо, що D - порядку m×k, F - порядку n×l, G i H - матриці одного порядку m×l.

 Зафіксуємо довільні i, j і доведемо, що gij = hij.Маємо

  k k k

 gij = ∑ diα cαj = ∑ ∑ aiβ bβα cαj ;

  α=1 α=1 β=1

  n n k

 hij = ∑ aiβfβj =∑ aiβ ∑ bβα cαj .

  β=1 β=1 α=1

 Позначивши tαβ = aiβ bβα cαj, отримаємо

 k n n k

 gij = ∑ ∑ tαβ , hij = ∑ ∑ tαβ .

 α=1 β=1 β=1 α=1

 Кожна із вказаних сум дорівнює сумі всіх елементів деякої матриці (tαβ), обчисленій двома різними способами. Отже, hij = gij, що й потрібно довести.

 Інші властивості добутку доводяться аналогіччно, тільки простіше.

 ***Оберненою*** називається матриця А-1, така що якщо її помножити на матрицю до якої вона обернена, то в результаті отримаємо одиничну матрицю. А\*А-1=Е

 Знайти матрицю, обернену до квадратної матриці М= аi k ,можна за допомогою операцій над розширеною матрицею А:

 m11 . . . . m1n  1 . . . . 0

 A= . . . . . . . . . . . . . . . .

 mn1 . . . . mnn 0 . . . . 1

 Якщо ліву частину матриці А звести елементарними перетвореннями до одиничної, то в правій частині дістанемо матрицю, обернену до М.

 До елементарних перетворень належать:

1)Переставлення двох рядків матриці А (або двох однойменних стовпців в лівій і правій частинах матриці А);

 2)Множення рядка на відмінне від нуля число( або однойменних стовпців в лівій і правій частинах матриці А);

 3)Заміна рядка сумою цього і будь-якого іншого рядка (або та ж сама сума однойменних стовпців в лівій і правій частинах матриці А);

 **Ділення двох матриць.**

 Дію ділення можна замінити дією множення на обернену матрицю

 \* В-1

PROGRAM povorot; {Поворот матриці }

 USES CRT;

 CONST

 N=3;

 TYPE

 S=ARRAY[1..N,1..N]OF REAL;

 SS=ARRAY[1..N,1..N]OF REAL;

 VAR

 S1:S;S2:SS;M,i,j:INTEGER;

 BEGIN

 FOR i:=1 TO N DO

 BEGIN

 FOR j:=1 TO N DO

 BEGIN

 READ(S1[i,j]); {Ввід матриці}

 END;

 END;

 WRITE('Vvedit kut povorotu');

 READ(M); {Ввід кута повороту}

 CASE M OF

 90:BEGIN {Поворот матриці на 90°}

 FOR i:=1 TO N DO

 FOR j:=1 TO N DO

 S2[I,J]:=S1[N-J+1,I];

 FOR i:=1 TO N DO

 FOR j:=1 TO N DO

 WRITELN(S2[i,j]);

 END;

 180: BEGIN {Поворот матриці на 180°}

 FOR I:=1 TO N DO

 FOR J:=1 TO N DO

 S2[I,J]:=S1[N-I+1,N-J+1];

 FOR I:=1 TO N DO

 FOR J:=1 TO N DO

 WRITELN(S2[I,J]);

 END;

 270: BEGIN {Поворот матриці на 270°}

 FOR I:=1 TO N DO

 FOR J:=1 TO N DO

 S2[I,J]:=S1[J,N-I+1];

 FOR I:=1 TO N DO

 FOR J:=1 TO N DO

 WRITELN(S2[I,J]); {Вивід результату}

 END; END;

 END.

 Program Suma; {Сума двох матриць}

 Const dim1=20;

 dim2=40;{dim2=2\*dim1}

 Type ar1=array[1..dim1,1..dim2] of real;

 ar2=array[1..dim1,1..dim2] of real;

 ar3=array[1..dim1,1..dim2] of real;

 Var i,j,n,m:integer;

 A:ar1;

 B:ar2;

 C:ar3;

 Begin

 write('Введіть розмірність матриці М');

 readln(n,m); {Ввід розмірності матриць }

 write('Введіть матрицю М');

 For i:=1 to n do

 For j:=1 to m do

 read(A[i,j]); {Ввід першої матриці}

 write('Введіть матрицю М');

 For i:=1 to n do

 For j:=1 to m do

 read(B[i,j]); {Ввід другої матриці}

 For i:=1 to n do

 For j:=1 to m do

 C[i,j]:=A[i,j]+B[i,j]; {Обчислення суми матриць}

 For i:=1 to n do

 For j:=1 to m do

 writeln(C[i,j]); {Вивід результату}

 End.

Результати:

 n=2,m=3

 1 -4 5 6 -1 0 7 -5 5

 0 3 8 6 0 -9 6 3 -1

Program Rizn; {Знаходження різниці двох матриць}

 Const dim1=20;

 dim2=40;

 Type ar1=array[1..dim1,1..dim2] of real;

 ar2=array[1..dim1,1..dim2] of real;

 ar3=array[1..dim1,1..dim2] of real;

 Var i,j,n,m:integer;

 A:ar1;

 B:ar2;

 C:ar3;

 Begin

 write('Введіть розмірність матриці М');

 readln(n,m); {Ввід розмірності матриць}

 write('Введіть матрицю М');

 For i:=1 to n do

 For j:=1 to m do

 read(A[i,j]); {Ввід першої матриці}

 write('Введіть матрицю М');

 For i:=1 to n do

 For j:=1 to m do

 read(B[i,j]); {Ввід другої матриці}

 For i:=1 to n do

 For j:=1 to m do

 C[i,j]:=A[i,j]-B[i,j]; {Знаходження їх різниці}

 For i:=1 to n do

 For j:=1 to m do

 writeln(C[i,j]); {Вивід результату}

 End.

Результати:

n=2,m=3

 1 -4 5 6 -1 0 -5 -3 5

 0 3 8 6 0 -9 -6 3 17

if k<>i then

 For j:=n1 downto 1 do

 a[k,j]:=a[k,j]-a[i,j]\*a[k,i];end;

 For i:=1 to n do

 For j:=1 to n do

 m[i,j]:=a[i,j+n];

 For i:=1 to n do

 For j:=1 to n do

 writeln(m[i,j]:6:2); {Вивід оберненої матриці}

 End.

 2 4 3 -0,2 0 0,2

 М= 0 -1 -5 М-1 = 0,41 0,18 -0,112

 7 4 3 -0,08 -0,24 0,02

Program Dobutok; {Множення числа на матрицю}

 Const dim1=20;

 dim2=40;{dim2=2\*dim1}

 Type ar1=array[1..dim1,1..dim2] of real;

 ar2=array[1..dim1,1..dim2] of real;

 Var i,j,n,m:integer;

 A:ar1; C:ar2;

 r:real;

 Begin

 write('Введіть число');

 readln(r); {Ввід числа}

 write('Введіть розмірність матриці М');

 readln(n,m); {Ввід розмірності матриці}

 write('Введіть матрицю М');

 For i:=1 to n do

 For j:=1 to m do

 read(A[i,j]); {Ввід матриці}

 For i:=1 to n do

 For j:=1 to m do

 C[i,j]:=A[i,j]\*r; {Множення матриці на число}

 For i:=1 to n do

 For j:=1 to m do

 writeln(C[i,j]); {Вивід результату}

 End.

Резуьтати:

r=5 n=3,m=2

 -1 2 -5 10

 M= 3 5 C= 15 25

 4 -2.5 20 -12.5

Program obernena; {Знаходження оберненої матриці}

 Const dim1=20;

 dim2=40;{dim2=2\*dim1}

 Type ar=array[1..dim1,1..dim1] of real;

 ar1=array[1..dim1,1..dim2] of real;

 Var i,j,k,n1:integer; n:integer;Var m:ar;e:real;

 t,s:real;

 a:ar1;

 Begin

 readln(e); {Ввід точності обчислень}

 write('Введіть розмірність матриці М');

 readln(n); {Ввфд розмірності квадратної матриці}

 write('Введіть матрицю М');

 For i:=1 to n do

 For j:=1 to n do

 read(m[i,j]); {Ввід матриці}

 n1:=2\*n;

 For i:=1 to n do

 For j:=1 to n1 do

 if j<=n then a[i,j]:=m[i,j]

 else if j=n+i then a[i,j]:=1

 else a[i,j]:=0;

 For i:=1 to n do

 begin k:=i;s:=a[i,i];

 for j:=i+1 to n do

 begin t:=a[j,i];

 if abs(s)<abs(t) then begin s:=t;k:=j;end;end;

 if abs(s)<e then begin {Знаходження визначника}

 write('M-вироджена');Halt end;

 For j:=i to n1 do

 begin t:=a[k,j];a[k,j]:=a[i,j];a[i,j]:=t/s;end; {Знаходження елементів оберненої }

 For k:=1 to n do {матриці}

рrgram dob\_mas;

 Const dim1=30;

 dim2=40;

 dim3=30;

 Type ar1=array[1..dim2,1..dim1] of real;

 ar2=array[1..dim3,1..dim2] of real;

 ar3=array[1..dim1,1..dim3] of real;

 Var i,j,k,l,n,m:integer;

 A:ar1;

 B:ar2;

 C:ar3;

 Begin

 write('Введіть розмірність матриці М');

 readln(n,m,l);

 write('Введіть матрицю А');

 For i:=1 to n do

 For j:=1 to m do

 read(A[i,j]);

 write('Введіть матрицю В');

 For i:=1 to l do

 For j:=1 to n do

 read(B[i,j]);

 For i:=1 to m do

 For j:=1 to l do

 For k:=1 to n do

 C[i,j]:=C[i,j]+A[k,i]\*B[j,k];

 For i:=1 to m do

 For j:=1 to l do

 writeln(C[i,j]);

 End.

 Результати:

n=2,m=2,l=2,

 1 2 3 4

 3 4 1 2

рrogram dilennya; {Ділення двох матриць}

 Const dim1=20;

 dim2=40;{dim2=2\*dim1}

 Type ar1=array[1..dim1,1..dim1] of real;

 ar2=array[1..dim1,1..dim1] of real;

 ar3=array[1..dim1,1..dim2] of real;

 ar4=array[1..dim1,1..dim2] of real;

 Var i,j,k,n1:integer; n:integer; M:ar3;e:real;

 t,s:real;

 A:ar1;B:ar2;c:ar4;

 Begin

 readln(e); {Знаходження оберненої матриці }

 write('Введіть розмірність матриці М');

 readln(n);

 write('Введіть матрицю M');

 For i:=1 to n do

 For j:=1 to n do

 read(M[i,j]);

 write('Введіть матрицю B');

 For i:=1 to n do

 For j:=1 to n do

 read(B[i,j]);

 n1:=2\*n;

 For i:=1 to n do

 For j:=1 to n1 do

 if j<=n then a[i,j]:=m[i,j]

 else if j=n+i then a[i,j]:=1

 else a[i,j]:=0;

 For i:=1 to n do

 begin k:=i;s:=a[i,i];

 for j:=i+1 to n do

 begin t:=a[j,i];

 if abs(s)<abs(t) then begin s:=t;k:=j;end;end;

 if abs(s)<e then begin

 write('M- вироджена');Halt end;

 For j:=i to n1 do

 begin t:=a[k,j];a[k,j]:=a[i,j];a[i,j]:=t/s;end;

 For k:=1 to n do

 if k<>i then

 For j:=n1 downto 1 do

 a[k,j]:=a[k,j]-a[i,j]\*a[k,i];end;

 For i:=1 to n do

 For j:=1 to n do

 m[i,j]:=a[i,j+n];

 For i:=1 to n do

 For j:=1 to n do

 For k:=1 to n do

 C[i,j]:=C[i,j]+B[k,i]\*M[j,k]; {Множення матриці на обернену}

 For i:=1 to n do

 For j:=1 to n do

 writeln(C[i,j]:6:2); {Вивід результату}

 End.

**Результати:**

**n=3**

 **1 2 3 1 2 3**

**M= 4 5 6 B= 4 5 6**

 **7 8 9 7 8 9**

 **1 0 0**

**C= 0 1 0**

 **0 0 1**

 ***Початок***

 **Ввід n,m**

 **i=1,n**

 **j=1,m**

 **Ввід А[i,j]**

 **i=1,n**

 **j=1,m**

 **Ввід B[i,j]**

 **i=1,n**

 **j=1,m**

**C[i,j]=A[i,j]+B[i,j]**

**i=1,n**

**j=1,m**

**Вивід С[i,j]**

**Кінець**

 ***Початок***

 **Ввід n,m**

 **i=1,n**

 **j=1,m**

 **Ввід А[i,j]**

 **i=1,n**

 **j=1,m**

 **Ввід B[i,j]**

 **i=1,n**

 **j=1,m**

**C[i,j]=A[i,j]-B[i,j]**

**i=1,n**

**j=1,m**

**Вивід С[i,j]**

**Кінець**

 ***Початок***

 **Ввід n,m,l**

 **i=1,n**

 **j=1,m**

 **Ввід А[i,j]**

 **i=1,l**

 **j=1,n**

 **Ввід B[i,j]**

 **i=1,m**

 **j=1,l**

 **k=1,n**

**C[i,j]=С[i,j]+А[k,i]\*В[j,k]**

**i=1,m**

**j=1,l**

**Вивід С[i,j]**

 ***Початок***

 **Ввід e,n**

 **i=1,n**

 **j=1,n**

 **Ввід M[i,j]**

**n1=2\*n**

 **i=1,n**

 **j=1,n**

так j=n ні

a[i,j]=m[i,j] так j=n+1 ні

 a[i,j]=1 a[i,j]=0

 **i=1,n**

k=i s=a[i,i]

 **j=1,n**

t=a[j,i]

 s < t так

s=t k=j

 s < e так М-вироджена

 **j=i,n1**

**t=a[k,j] a[k,j]=a[i,j] a[i,j]=t/s**

 **k=1,n**

**k<>i**

 **j=n1,1**

**a[k,j]=a[k,j]-a[i,j]\*a[k,i]**

 **i=1,n**

 **j=1,n**

**M[i,j]=A[i,j=m]**

**i=1,n**

**j=1,n**

**Вивід M[i,j]**

**Кінець**

 ***Початок***

 **Ввід e,n**

 **i=1,n**

 **j=1,n**

 **Ввід M[i,j]**

**n1=2\*n**

 **i=1,n**

 **j=1,n**

так j=n ні

a[i,j]=m[i,j] так j=n+1 ні

 a[i,j]=1 a[i,j]=0

 **i=1,n**

k=i s=a[i,i]

 **j=1,n**

t=a[j,i]

 s < t так

s=t k=j

 s < e так М-вироджена

 **j=i,n1**

**t=a[k,j] a[k,j]=a[i,j] a[i,j]=t/s**

 **k=1,n**

**k<>i**

 **j=n1,1**

**a[k,j]=a[k,j]-a[i,j]\*a[k,i]**

 **i=1,n**

 **j=1,n**

**M[i,j]=A[i,j=m]**

 **i=1,n**

 **j=1,n**

 **k=1,n**

**C[i,j]=С[i,j]+B[k,i]\*M[j,k]**

**i=1,n**

**j=1,n**

**Вивід С[i,j]**

**Кінець**

 ***Початок***

 **Ввід n**

 **i=1,n**

 **j=1,n**

 **Ввід S1[i,j]**

**Ввід М**

**М=90° M=180° M=270°**

 **i=1,n i=1,n i=1,n**

 **j=1,n j=1,n j=1,n**

**S2[i,j]=S1[n-j+1,i] S2[i,j]=S1[n-i+1,n-j+1] S2]i,j]=S1[j,n-i+1]**

**i=1,n**

**j=1,n**

**Вивід S2[i,j]**

**Кінець**

Program kursov;

type E=ARRAY [1..3,1..3] of real;

var A,B,C,S:E;

 f,a1,b1:real;

 t:integer;

 mass:char;

 procedure INPUT (mass:char; var A:E);

 var x,y,n:integer;

 begin

 writeln ('‚ўi¤ Ґ«Ґ¬Ґ­вiў ¬ бЁўг ',mass);

 for y:=1 to 3 do

 begin

 for x:=1 to 3 do

 read (A[x,y]);

 end;

 end;

 procedure SUMM (A,B:E; var S:E);

 var x,y:integer;

 begin

 for y:=1 to 3 do

 for x:=1 to 3 do

 S[x,y]:=A[x,y]+B[x,y];

 end;

 procedure SUB (A,B,S:E; var C:E;b2:real);

 var x,y,i:integer;

 begin

 SUMM (A,B,S);

 write ('Њ бЁў C');

 for y:=1 to 3 do

 begin

 writeln;

 for x:=1 to 3 do

 begin

 C[x,y]:=0;

 for i:=1 to 3 do

 C[x,y]:=C[x,y]+(S[i,x]\*B[i,y]);

 write (C[x,y]:7:2);

 if (x=1) AND (y=1) then b2:=C[x,y]

 else

 if C[x,y]<b2 then b2:=C[x,y];

 end;

 end;

 writeln;

 writeln ('Њi­i¬ «м­Ё© Ґ«Ґ¬Ґ­в ¬ ваЁжi ‘ у ',b2:7:2);

 end;

 begin

 mass:='A';

 INPUT (mass,A);

 mass:='B';

 INPUT (mass,B);

 SUB (A,B,S,C,b1);

 writeln (' ’ Ў«Ёжп §­ зҐ­м ¤«п § ¤ ­®х дг­Єжiх');

 writeln (' t f');

 for t:=1 to 50 do

 begin

 if t>40 then a1:=0.2

 else

 a1:=0.11;

 f:=SIN(a1\*t)-EXP(2\*a1\*t)+b1;

 writeln(t:8,f:16:4);

 end;

 end.

 ***Початок***

 **Ввід n,m**

 **i=1,n**

 **j=1,m**

 **Ввід А[i,j]**

 **i=1,n**

 **j=1,m**

 **Ввід B[i,j]**

 **i=1,n**

 **j=1,m**

**S[i,j]=A[i,j]+B[i,j]**

 **i=1,m**

 **j=1,l**

 **k=1,n**

**C[i,j]=С[i,j]+А[k,i]\*В[j,k]**

**i=1,n**

**j=1,m**

**Вивід С[i,j]**

 **так х=1,y=1 ні**

 **b2=C[x,y]**

**так C[x,y]<b2 ні**

 **b2=C[x,y]**

**Вивід b2**

**Кінець**

**Задача.**

**Надрукувати порядково матрицю С і її мінімальний елемент.**

 **Для вирахування матриці С використати стандартну підпрограму додавання матриць і складену підпрограму множення матриць.**

 **1,2 0,7 2,6 4,5 -6,8 -1,2**

**А= 3,3 0,4 2,0 В= 3,8 0,5 0,8**

 **1,7 1,5 -0,2 1,2 0,6 0,25**

 **-30,61 23,73 10,62**

**С= -36,09 -21,05 -6,26**

 **-12,68 6,68 3,34**