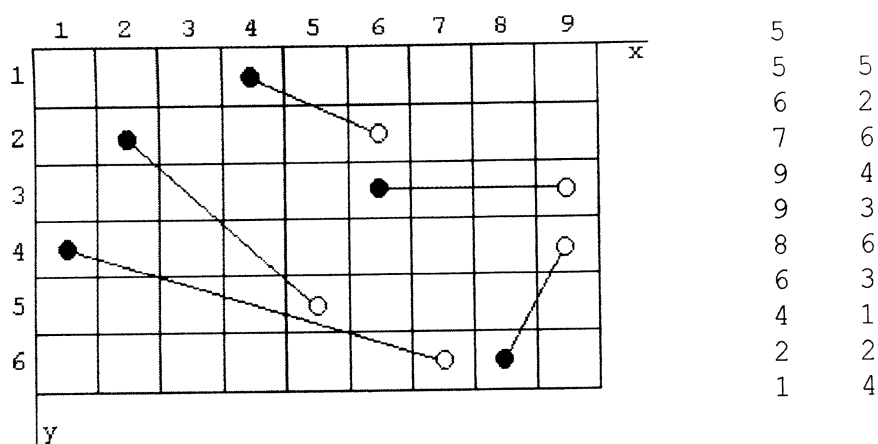


O placă dreptunghiulară de dimensiuni  $L1 \times L2$  ( $L1 \leq 600$ ,  $L2 \leq 300$ ) este împărțită într-o rețea ortogonală de pătrate de latură 1, în care sunt marcate  $N$  puncte albe și  $N$  puncte negre, toate având coordonatele întregi. Oricare 3 din cele  $2N$  puncte nu sunt coliniare. În scopul obținerii unui circuit imprimat prin corodarea plăcii, fiecare punct alb urmează a fi unit printr-un segment de dreaptă cu un punct negru oarecare, dar astfel încât cele  $N$  segmente să nu se intersecteze.

Programul trebuie să realizeze și o verificare a necoliniarității punctelor de pe placă. Dacă printre cele  $2N$  puncte se găsesc 3 coliniare, se va tipări mesajul "Wrong Input" și executarea programului se va încheia.



Datele de intrare se furnizează dintr-un fișier text (placa.in), cu următorul conținut:

- pe prima linie apare valoarea lui  $N$ ;
- pe următoarele  $2N$  linii apar două numere naturale, reprezentând coordonatele punctelor albe, urmate de coordonatele punctelor negre.

Pentru exemplul considerat, conținutul fișierului de intrare apare mai sus în dreapta figurii. Rezultatele vor fi tipărite în alt fișier text (placa.out) pe  $N$  linii, fiecare linie conținând numărul de ordine al unui punct alb, urmat de numărul de ordine al punctului negru cu care este conectat.

Pentru desenul din figura de mai sus, o soluție posibilă este:

```

1  4
2  3
3  5
4  1
5  2

```

### Problema 3- Navigare pe Web(100 puncte)

Se considera  $n$  site-uri Web complet interconectate (fiecare site are câte un link la fiecare din celelate site-uri). Un hacker plictisit, dorește să viziteze toate cele  $n$  site-uri, iar fiecare site e vizitat o singură dată. După ce a vizitat cele  $n$  site-uri, hacker-ul dorește să ajunga iarăși la site-ul de la care a început vizitarea. În câte moduri poate vizita hacker-ul cele  $n$  site-uri, pornind de pe un anumit site stabilit, în condițiile menționate mai sus.

In fisierul de intrare WEB.IN este scris numarul n.  
In fisierul de iesire WEB.OUT se va scrie numarul de moduri de vizitare a celor n site-uri.

Limite:

- $2 < n \leq 100$ ;
- site-ul de la care porneste hacker-ul e singurul site vizitat de doua ori.

Exemplu:

WEB.IN	WEB.OUT
4.	3

Observatie: Se cere numarul de moduri distincte de a vizita cele n site-uri.

De exemplu, pentru  $n=4$  vizitarea paginilor in ordinea 1-2-3-4-1 este identica cu vizitarea paginilor in ordinea 1-4-3-2-1.

Timp de executie: 1 secunda/test. Punctajul 100 puncte.

Fisier sursa	Fisier de intrare	Fisier de iesire
WEB.PAS sau	WEB.IN	WEB.OUT
WEB.CPP		

Date de Test:

	WEB.IN	WEB.OUT
1.	3	1
2.	8	2520
3.	10	181440
4.	20	60822550204416000
5.	45	55 cifre=> 132913578739422438402181290550730794...6400000000000
6.	50	63 cifre=> 304140932...605120000000000
7.	77	111 cifre=> 942747350...861120000000000000000000
8.	83	123 cifre=> 237682166...559680000000000000000000
9.	90	136 cifre=> 825397758...987840000000000000000000
10.	100	156 cifre=> 466631077...584320000000000000000000

Observatii:

- 1 test = 10 puncte
- exista 10 teste pentru fiecare problema
- punctajul total = 300 puncte
- nu exista puncte din oficiu.

Se dau două şiruri de numere întregi  $a(i)$ ,  $i=1,n$  şi  $b(j)$ ,  $j=1,m$ . Se cere să se afişeze cel mai lung subşir comun ordonat crescător.

Datele se citesc din fişier.

Fisierul de intrare: intrare.in  
n,m

sir 1  
sir 2

fișierul de ieșire : iesire.out contine subsirul cerut

Timp de execuție 1 sec.

Date de test:

Set 1:

Intrare.in:

7 5  
8 0 -3 4 -4 9 10  
2 0 4 9 10

Set 2:

Intrare.in:

5 3  
1 0 2 0 3  
6 7 8

Set 3:

Intrare.in:

30 28  
1 0 2 0 3 4 5 6 0 7 0 8 9 10 0 11 12 13 14 15 16 17 18 19 20 21 22 23 24 0 25  
1 2 3 4 5 6 7 8 9 10 11 0 12 0 13 14 15 16 17 18 19 20 21 0 22 0 23 24