

Tuyển tập đề thi tin học quốc gia (2005-2008)

Đề thi vòng I quốc gia

Năm 2005

Bảng A

Bài 1. Phân đoạn

Tên file chương trình: **SEGPAS.PAS**

Cho dãy số nguyên a_1, a_2, \dots, a_n và số nguyên dương k . Ta gọi k -phân đoạn của dãy số đã cho là cách chia dãy số đã cho ra thành k đoạn, mỗi đoạn là một dãy con gồm các phần tử liên tiếp của dãy. Chính xác hơn, một k -phân đoạn được xác định bởi dãy chỉ số

$$1 \leq n_1 < n_2 < \dots < n_k = n.$$

Đoạn thứ i là dãy con $a_{n_{i-1}+1}, a_{n_{i-1}+2}, \dots, a_{n_i}, i = 1, 2, \dots, k$. Ở đây quy ước $n_0 = 0$.

Yêu cầu: Hãy xác định số M nhỏ nhất để tồn tại k -phân đoạn sao cho tổng các phần tử trong mỗi đoạn đều không vượt quá M .

Dữ liệu: Vào từ file văn bản SEGPAS.INP.

- Dòng đầu tiên chứa hai số nguyên n và k ($1 \leq k \leq n \leq 15000$);
- Dòng thứ i trong số n dòng tiếp theo chứa số nguyên a_i ($|a_i| \leq 30000$), $i = 1, 2, \dots, n$.

Các số cạnh nhau trên một dòng trong file dữ liệu cách nhau ít nhất một dấu cách.

Kết quả: Ghi ra file SEGPAS.OUT một số nguyên duy nhất là giá trị M tìm được.

Ví dụ:

SEGPAS.INP	
9	4
1	
1	
1	
3	
2	
2	
1	
3	
1	

SEGPAS.OUT
5

Bài 2: Pháo hoa

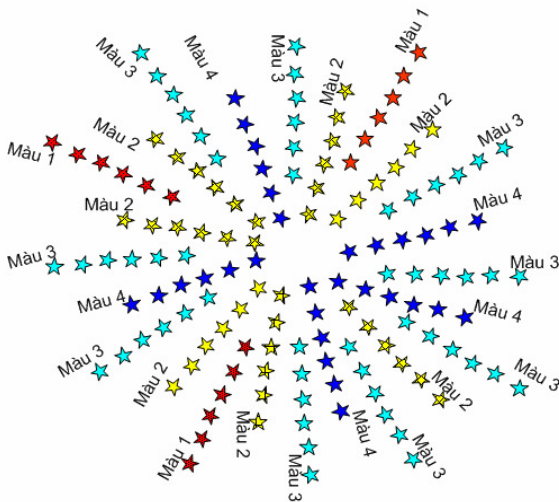
Tên chương trình: FIREWK.PAS

Nhằm chào mừng các ngày lễ lớn trong năm 2005 người ta đã chế tạo một loại đạn pháo hoa mới, khi bắn, đạn nổ thành bông hoa $2n$ cánh màu ($1 \leq n \leq 30$). Nguyên vật liệu cho phép tạo được m màu khác nhau, đánh số từ 1 đến m ($2 \leq m \leq 32$).

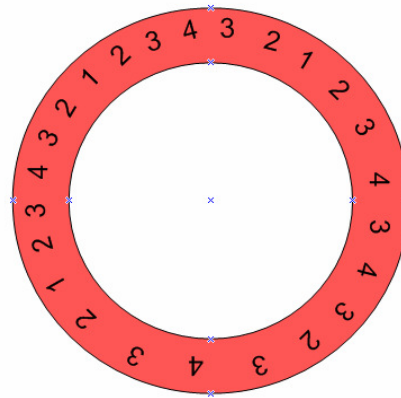
Để đảm bảo tính mỹ thuật, việc chuyển tiếp màu giữa 2 cánh hoa kề nhau phải tuân theo quy tắc chuyển màu cầu vồng sau đây:

- Bên cạnh cánh hoa màu i phải là cánh hoa màu $i-1$ hoặc $i+1$, với $1 < i < m$,
- Bên cạnh cánh hoa màu 1 chỉ có thể là cánh hoa màu 2,
- Bên cạnh cánh hoa màu m chỉ có thể là cánh hoa màu $m-1$.

Một bông hoa không nhất thiết phải có đầy đủ m màu. Mỗi bông hoa tương ứng với một vòng tròn $2n$ số thể hiện màu của các cánh hoa. Ví dụ, hình 1 là bông hoa 24 cánh ($n = 12$) và hình 2 là vòng tròn 9 số



Hình 1



Hình 2

tương ứng với nó. Mỗi bông hoa được mô tả bằng dãy $2n$ số nguyên liệt kê các chỉ số màu của các cánh hoa theo chiều kim đồng hồ. Ví dụ, bông hoa ở hình 1 có thể được mô tả bằng dãy số

3 4 3 2 1 2 3 4 3 2 1 2 3 4 3 2 1 2 3 4 3 4 3 2.

Dãy có thứ tự từ điển nhỏ nhất trong các dãy có thể dùng để mô tả hoa được gọi là mã hoa. Khi đó, mã hoa của bông hoa ở hình 1 sẽ là

1 2 3 4 3 2 1 2 3 4 3 2 1 2 3 4 3 4 3 2 3 4 3 2.

Trong các ngày lễ, Ban tổ chức yêu cầu bắn các đạn pháo hoa $2n$ cánh có đúng k cánh màu C ($0 \leq k \leq 2$). Các mã hoa thỏa mãn yêu cầu vừa nêu cũng được sắp xếp theo thứ tự từ điển và đánh số bắt đầu từ 1. Hơn nữa, nhằm tạo ra các hoa không giống nhau, đội bắn pháo hoa cần đảm bảo hai viên đạn pháo hoa bắn liên tiếp phải có mã khác nhau. Do vậy, người ta đã thiết kế một Hệ thống chụp ảnh và phân tích tự động để báo cho đội bắn pháo hoa biết số thứ tự của viên đạn pháo hoa vừa nổ trên trời. Em được giao viết chương trình giải quyết nhiệm vụ chính trong phần mềm phân tích tự động này.

Yêu cầu: Cho biết n, m, k và C . Gọi X là tập tất cả các mã hoa $2n$ cánh có đúng k cánh màu C .

- Hãy xác định số lượng p các phần tử của X ;
- Cho một mã hoa nào đó trong tập X . Hãy xác định số thứ tự từ điển của nó trong X .

Dữ liệu: Vào từ file văn bản FIREWK.INP. Dòng đầu tiên chứa 4 số nguyên n, m, k, C ; dòng tiếp theo chứa $2n$ số nguyên mô tả một mã hoa.

Các số trên một dòng của file dữ liệu cách nhau ít nhất một dấu cách.

Kết quả: Đưa ra file văn bản FIREWK.OUT. Dòng đầu tiên ghi số nguyên p ; dòng tiếp theo ghi số thứ tự tìm được của mã hoa.

Ví dụ:

FIREWK.INP			
3	4	0	1
2	3	4	3 4 3

FIREWK.OUT
4
3

Bài 3. Bộ sưu tập

Tên chương trình: COLLECT.PAS

Một bộ sưu tập tiền xu cổ được coi là có giá trị phải gồm không ít hơn Z_0 đồng tiền vàng, S_0 đồng tiền bạc và M_0 đồng tiền đồng. Bộ sưu tập ban đầu của Alibaba có một số lượng nhất định các đồng tiền vàng, bạc và đồng nhưng chưa phải là một bộ sưu tập có giá trị. Tại Trụ sở của Hiệp hội những người sưu tầm tiền cổ có đặt một máy đổi tiền để giúp hội viên đổi được các bộ sưu tập có giá trị. Tuy nhiên, máy đổi chỉ hỗ trợ việc đổi tiền trọn gói theo quy tắc đổi gói (Z_1, S_1, M_1) lấy gói (Z_2, S_2, M_2) đồng tiền. Các quy tắc đổi tiền khác nhau từng đôi một, được gán số hiệu tuần tự 1,2,3, . . . và được công bố trước. Hội viên có thể tạo gói tiền thích hợp từ bộ sưu tập của mình để thực hiện việc đổi tiền. Các đồng tiền nhận được sau mỗi lần đổi được gộp lại với các đồng tiền mà hội viên đang có để thành *một bộ sưu tập mới* và có thể được sử dụng để đổi trong những lần sau nếu cần. Số lần đổi không hạn chế, tuy nhiên, là người thực dụng, *Alibaba luôn cố gắng giảm tới mức tối đa số lần đổi tiền*. Mặt khác, để ngăn chặn việc đầu cơ, Hiệp hội quy định, trong mọi thời điểm, mỗi hội viên không được giữ quá 4 đồng tiền mỗi loại và không được phép đổi tiếp khi đã đổi được một bộ sưu tập có giá trị.

Yêu cầu: Cho biết số lượng Z, S, M các đồng tiền vàng, bạc, đồng mà Alibaba có ban đầu và các quy tắc đổi tiền. Hãy chỉ ra tất cả các bộ sưu tập tiền cổ có giá trị mà Alibaba có thể có được sau một số lần đổi không vượt quá k cho trước.

Dữ liệu: Vào từ file văn bản COLLECT.INP.

- Dòng đầu ghi số nguyên dương $k(k \leq 1000)$; dòng thứ 2 ghi 6 số nguyên không âm $Z, S, M, Z_0, S_0, M_0 (0 \leq Z, S, M, Z_0, S_0, M_0 \leq 4)$;
- Các dòng tiếp theo mỗi dòng ghi 6 số nguyên không âm $Z_1, S_1, M_1, Z_2, S_2, M_2$ xác định một quy tắc đổi tiền $(0 \leq Z_1, Z_2, S_1, S_2, M_1, M_2 \leq 4)$.

Kết quả: Đưa ra file văn bản COLLECT.OUT. Nếu không tồn tại cách đổi để có được bộ sưu tập có giá trị, file kết quả chỉ gồm một số -1. Trong trường hợp ngược lại, dòng đầu ghi số v là số các bộ tiền cổ có giá trị mà Alibaba có thể đổi được. Dòng thứ i trong v dòng tiếp theo ghi 4 số nguyên Z_i, S_i, M_i, k_i mô tả bộ sưu tập có giá trị thứ i và số lần đổi k_i ít nhất không vượt quá k cần thực hiện để có được bộ sưu tập ấy.

Các số trên một dòng của file dữ liệu và file kết quả đặt cách nhau ít nhất một dấu cách.

Ví dụ:

COLLECT . INP						
2						
4	0	1	3	3	3	
1	0	1	1	1	1	
2	0	1	1	3	3	

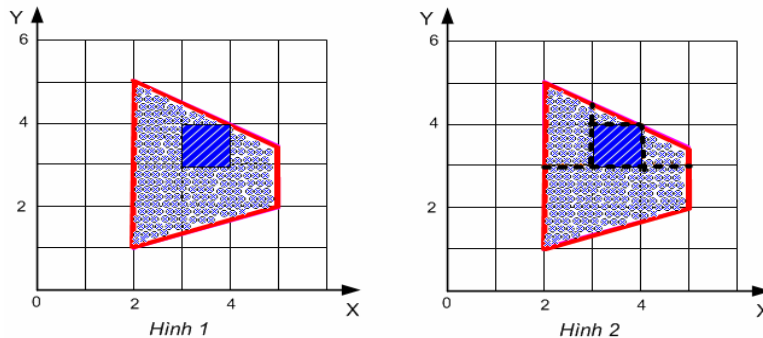
COLLECT . OUT				
1				
3	3	3	1	

Bài 4. Khuôn thép

Tên chương trình: STEEL.PAS

Để chuẩn bị cho Lễ hội kỷ niệm 30 năm ngày Chiến dịch Hồ Chí Minh toàn thắng, giải phóng miền Nam, thống nhất đất nước, người ta cần gia công các loại khuôn thép có hình dạng là các hình đa giác lồi M đỉnh. Mỗi khuôn thép được thiết kế trên một tấm thép cũng có hình dạng là một hình đa giác lồi N đỉnh, không có cạnh nào của khuôn thép nằm gọn trên một cạnh của tấm thép. Để tiện cho việc gia công, khuôn thép được vẽ sao cho hai đường thẳng chứa hai cạnh không kề nhau của nó không cắt nhau ở bên trong tấm thép.

Công việc chính cần làm trong quá trình gia công là sử dụng máy cắt để cắt được khuôn thép từ tấm thép ra. Rõ ràng là cần phải thực hiện M nhát cắt. Mỗi nhát cắt được thực hiện bằng cách chọn một cạnh nào đó của khuôn thép và cắt theo đường thẳng chứa cạnh ấy chia tấm thép thành hai phần, một phần chứa khuôn thép cần gia công. Chi phí cắt khuôn thép là tổng chiều dài của các đường cắt.



Trên hình 1 và 2, tấm thép là tứ giác được tô nhạt, khuôn thép là hình vuông được tô bằng các gạch đậm. Các nét gạch đứt là các đường cắt với tổng chi phí bằng 6.5 đơn vị.

Yêu cầu: Cho biết hình dạng tấm thép và khuôn thép cần gia công. Hãy tìm phương án cắt khuôn thép có chi phí nhỏ nhất.

Dữ liệu: Vào từ file văn bản STEEL.INP: Dòng đầu ghi số $N(3 \leq N \leq 2000)$ là số đỉnh của tấm thép; N dòng tiếp theo, mỗi dòng ghi 2 số thực x và y ($-10^4 < x, y < 10^4$), là tọa độ N đỉnh của tấm thép được liệt kê theo chiều kim đồng hồ bắt đầu từ một đỉnh nào đó; dòng tiếp theo ghi số $M(3 \leq M \leq 2000)$ là số đỉnh của khuôn thép; cuối cùng là M dòng, mỗi dòng ghi 2 số thực x và y ($-10^4 < x, y < 10^4$) là tọa độ M đỉnh của khuôn thép được liệt kê theo chiều kim đồng hồ bắt đầu từ một đỉnh nào đó. Các số trên một dòng cách nhau ít nhất một dấu cách.

Kết quả: Đưa ra file văn bản STEEL.OUT chi phí nhỏ nhất tìm được với độ chính xác tới 4 chữ số sau dấu chấm thập phân.

Ví dụ:

STEEL . INP	
4	
2	1
2	5
5	3 . 5
5	2
4	
3	3
3	4
4	4
4	3

STEEL . OUT
6 . 5000

Bảng B**Bài 1. Đóng gói sản phẩm***Tên file chương trình: ZXY.PAS*

Ở đầu ra của một dây chuyền sản xuất trong nhà máy ZXY có một máy xếp tự động. Sau khi kết thúc việc gia công trên dây chuyền, các sản phẩm sẽ được xếp vào các hộp có cùng dung lượng M . Sản phẩm rời khỏi dây chuyền được xếp vào hộp đang mở (khi bắt đầu ca làm việc có một hộp rỗng được mở sẵn) nếu như dung lượng của hộp còn đủ để chứa sản phẩm. Trong trường hợp ngược lại, máy sẽ tự động đóng nắp hộp hiện tại, cho xuất xưởng rồi mở một hộp rỗng mới để xếp sản phẩm vào. Trong một ca làm việc có n sản phẩm đánh số từ 1 đến n theo đúng thứ tự mà chúng rời khỏi dây chuyền. Sản phẩm thứ i có trọng lượng là a_i , $i = 1, 2, \dots, n$. Ban Giám đốc nhà máy qui định rằng sản phẩm xuất xưởng của mỗi ca làm việc phải được xếp vào trong không quá k hộp.

Yêu cầu: Hãy giúp người quản đốc của ca làm việc xác định giá trị M nhỏ nhất sao cho số hộp mà máy tự động cần sử dụng để xếp đầy n sản phẩm xuất xưởng của ca không vượt quá số k cho trước.

Dữ liệu: Vào từ file văn bản ZXY.INP:

- Dòng đầu tiên chứa hai số nguyên n và k , ($1 \leq k \leq n \leq 15000$);
- Dòng thứ i trong n dòng tiếp theo chứa số nguyên dương a_i ($a_i \leq 30000$), $i = 1, 2, \dots, n$.

Các số trên một dòng cách nhau ít nhất một dấu cách.

Kết quả: Ghi ra file ZXY.OUT một số nguyên duy nhất là dung lượng của hộp.

Ví dụ:

ZXY . INP	ZXY . OUT
9 4	5
1	
1	
1	
3	
2	
2	
1	
3	
1	

Bài 2. Chữ số**Tên chương trình: DIGIT.PAS**

Cho chuỗi M không quá 127 ký tự lấy từ tập $F = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F\}$ và không bắt đầu bằng ký tự 0. Gọi S là chuỗi với giá trị ban đầu là chuỗi M .

Người ta biến đổi M theo quy tắc sau: đếm số lần xuất hiện các ký tự $0, 1, 2, \dots, F$, gọi K_i là số lần xuất hiện ký tự i (với i lần lượt là $0, 1, 2, \dots, F$). Với các $K_i \neq 0$ người ta viết liên tiếp chuỗi biểu diễn số K_i trong cơ số 16 và ký tự i . Chuỗi kết quả thu được là giá trị mới của M . Sau mỗi lần biến đổi người ta lại viết tiếp M vào sau S .

Ví dụ, với $M = '150A'$, S nhận giá trị ban đầu là $'150A'$.

Sau lần biến đổi thứ nhất ta có M là $'1011151A'$ và $S = '150A1011151A'$.

Sau lần biến đổi thứ 2 ta có M là $'1051151A'$ và $S = '150A1011151A1051151A'$.

Sau lần biến đổi thứ 3 ta có M là $'1041251A'$ và $S = '150A1011151A1051151A1041251A'$.

Yêu cầu: Cho chuỗi M , số lần biến đổi L ($0 \leq L \leq 10^7$) và X là một ký tự từ tập F . Hãy đếm số lần xuất hiện X trong S thu được sau L lần biến đổi M .

Dữ liệu: Vào từ file văn bản DIGIT.INP :

- Dòng thứ nhất chứa chuỗi M ,
- Dòng thứ 2 chứa số nguyên L
- Dòng thứ 3 chứa ký tự X .

Kết quả: Đưa ra file văn bản DIGIT.OUT một số nguyên - số lần xuất hiện X .

Ví dụ:

DIGIT . INP
150A
3
2

DIGIT . OUT
1

Bài 3. Đồi đất

Tên file chương trình: LAND.PAS

Hai bộ lạc Anpha và Bêta sống rất hoà thuận với nhau. Một phần ranh giới của hai bộ lạc là một đường gấp khúc không tự cắt. Đường gấp khúc nhận được bằng cách lần lượt nối N điểm đôi một khác nhau A_1, A_2, \dots, A_N . Điểm A_i được xác định bởi hoành độ x_i và tung độ y_i (x_i là các số nguyên thoả mãn điều kiện: $x_i \leq x_{i+1}$). Phần đất của bộ lạc Anpha nằm ở phía trên đường gấp khúc.

Nhân dịp năm mới, tù trưởng hai bộ lạc quyết định thay đường ranh giới cũ bằng cách xây dựng một đường cao tốc nối thẳng từ A_1 tới A_N và lấy đường cao tốc này ranh giới mới. Dĩ nhiên, sự thay đổi này sẽ chuyển một phần đất của bộ lạc Anpha cho bộ lạc Bêta và ngược lại. trướng thoả thuận phần diện tích chênh lệch do việc thay đường ranh giới sẽ được điều chỉnh trong tương lai bằng cách khác.

Yêu cầu: Hãy tính diện tích phần đất SA của bộ lạc trở thành đất của bộ lạc Bêta và diện tích phần đất SB của bộ lạc Bêta trở thành đất của bộ lạc Anpha sau thay đổi đường ranh giới giữa hai bộ lạc.

Dữ liệu: Vào từ file văn bản LAND.INP trong đó :

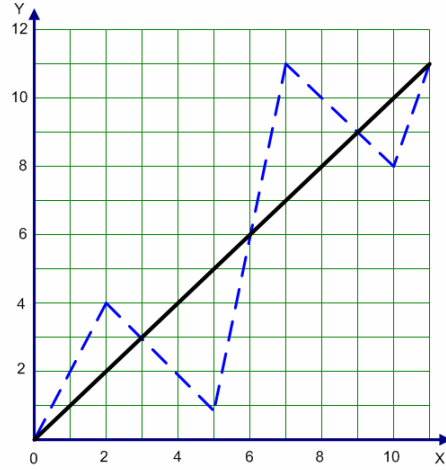
- Dòng đầu chứa số $N(N \leq 10000)$;
- Dòng thứ i trong N dòng tiếp theo chứa hai số nguyên x_i và y_i đặt cách nhau ít nhất 1 dấu cách ($-32000 \leq x_i, y_i \leq 32000$).

Kết quả: Đưa ra file văn bản LAND.OUT trong đó dòng thứ nhất chứa SA , dòng thứ hai chứa SB . Kết quả được lấy chính xác với 4 chữ số sau dấu chấm thập phân.

Ví dụ :

LAND . INP	
6	
0	0
2	4
5	1
7	11
10	8
11	11

LAND . OUT	
8.0000	
9.0000	



đôi
tốc là
làm
số
Hai từ
đổi
một
Anpha
khi

Bài 4. Bộ sưu tập

Tên chương trình: COLLECT.PAS

Một bộ sưu tập tiền xu cổ được coi là có giá trị phải gồm không ít hơn Z_0 đồng tiền vàng, S_0 đồng tiền bạc và M_0 đồng tiền đồng. Bộ sưu tập ban đầu của Alibaba có một số lượng nhất định các đồng tiền vàng, bạc và đồng nhưng chưa phải là một bộ sưu tập có giá trị. Tại Trụ sở của Hiệp hội những người sưu tầm tiền cổ có đặt một máy đổi tiền để giúp hội viên đổi được các bộ sưu tập có giá trị. Tuy nhiên, máy đổi chỉ hỗ trợ việc đổi tiền trọn gói theo quy tắc đổi gói (Z_1, S_1, M_1) lấy gói (Z_2, S_2, M_2) đồng tiền. Các quy tắc đổi tiền khác nhau từng đôi một, được gán số hiệu tuần tự 1,2,3, . . . và được công bố trước. Hội viên có thể tạo gói tiền thích hợp từ bộ sưu tập của mình để thực hiện việc đổi tiền. Số lần đổi tiền là không hạn chế, tuy nhiên, để ngăn chặn việc đầu cơ, Hiệp hội quy định mỗi hội viên không được giữ quá 4 đồng tiền mỗi loại. Các đồng tiền nhận được sau mỗi lần đổi được gộp lại với các đồng tiền mà hội viên đang có để thành một bộ sưu tập mới và có thể được sử dụng để đổi trong những lần sau nếu cần.

Yêu cầu: Cho biết số lượng Z, S, M các đồng tiền vàng, bạc, đồng mà Alibaba có ban đầu và các quy tắc đổi tiền. Hãy chỉ ra một phương án đổi tiền nào đó để Alibaba có được bộ sưu tập có giá trị. Dữ liệu vào đảm bảo luôn có phương án.

Dữ liệu: Vào từ file văn bản COLLECT.INP:

- Dòng đầu ghi 6 số nguyên không âm $Z, S, M, Z_0, S_0, M_0 (0 \leq Z, S, M, Z_0, S_0, M_0 \leq 4)$;
- Các dòng tiếp theo mỗi dòng ghi 6 số nguyên không âm $Z_1 S_1 M_1 Z_2 S_2 M_2$ xác định một quy tắc đổi tiền.

Kết quả: Đưa ra file văn bản COLLECT.OUT một dòng ghi dãy số hiệu các quy tắc theo thứ tự đã sử dụng trong phương án đổi tiền.

Các số trên một dòng của file dữ liệu và file kết quả đặt cách nhau ít nhất một dấu cách.

Ví dụ:

COLLECT . INP					
4	0	1	3	3	3
1	0	1	0	2	2
0	1	1	0	0	3
2	0	1	1	2	3
1	0	0	1	1	0

COLLECT . OUT	
3	4

Năm 2006

Bảng A

Bài 1. Dãy con dài nhất

Cho dãy số nguyên

$$a_1, a_2, \dots, a_n.$$

Dãy số

$$a_i, a_{i+1}, \dots, a_j$$

với $1 \leq i \leq j \leq n$ được gọi là dãy con của dãy số đã cho và khi đó, $j-i+1$ được gọi là **độ dài**, còn $\sum_{k=i}^j a_k$ được gọi là **trọng lượng** của dãy con này.

Yêu cầu: Cho số nguyên p , trong số các dãy con của dãy số đã cho có trọng lượng không nhỏ hơn p hãy tìm dãy con có độ dài lớn nhất.

Dữ liệu: Vào từ file văn bản MAXSEQ.INP:

- Dòng đầu tiên ghi hai số nguyên n và p cách nhau bởi dấu cách;
- Dòng thứ i trong số n dòng tiếp theo chứa số nguyên a_i là số hạng thứ i của dãy số đã cho, $i = 1, 2, \dots, n$.

Kết quả: Ghi ra file văn bản MAXSEQ.OUT số nguyên k là độ dài của dãy con tìm được (qui ước: nếu không có dãy con nào thoả mãn điều kiện đặt ra thì $k = -1$).

Ví dụ:

MAXSEQ . INP	MAXSEQ . OUT
5 6 -2 3 2 -2 3	4

MAXSEQ . INP	MAXSEQ . OUT
4 9 2 3 2 -2	-1

Hạn chế: Trong tất cả các test: $1 \leq n \leq 20000$; $|a_i| \leq 20000$; $|p| \leq 10^9$. Có 50% số lượng test với $n \leq 1000$.

Bài 2. Đường đi trên lưới

Cho một lưới ô vuông gồm m dòng và n cột. Các dòng được đánh số từ 1 đến m từ trên xuống dưới, các cột được đánh số từ 1 đến n từ trái qua phải. Ô nằm ở vị trí dòng i và cột j của lưới được gọi là ô (i,j) và khi đó, i được gọi là toạ độ dòng còn j được gọi là toạ độ cột của ô này. Trên ô (i,j) của lưới ghi số nguyên dương a_{ij} , $i = 1, 2, \dots, m$; $j = 1, 2, \dots, n$. Trên lưới đã cho, từ ô (i,j) ta có thể di chuyển đến ô (p,q) nếu các điều kiện sau đây được thoả mãn:

- $j < n$; $i \leq p$; $j \leq q$ và $i + j < p + q$;
- a_{ij} và a_{pq} có ước số chung lớn hơn 1.

Ta gọi một cách di chuyển từ mép trái sang mép phải của lưới là cách di chuyển bắt đầu từ một ô có toạ độ cột bằng 1 qua các ô của lưới tuân theo qui tắc di chuyển đã nêu và kết thúc ở một ô có toạ độ cột bằng n .

Yêu cầu: Tính số cách di chuyển từ mép trái sang mép phải của lưới.

Dữ liệu: Vào từ file văn bản NETPATH.INP:

- Dòng đầu tiên ghi 2 số nguyên dương m, n .
- Dòng thứ i trong số m dòng tiếp theo ghi n số nguyên dương $a_{i1}, a_{i2}, \dots, a_{in}$ là các số trên dòng thứ i của lưới, $i = 1, 2, \dots, m$.

Hai số liên tiếp trên cùng một dòng được ghi cách bởi ít nhất một dấu cách.

Kết quả: Ghi ra file văn bản NETPATH.OUT số nguyên k là số lượng cách di chuyển tìm được, biết rằng dữ liệu đảm bảo $k < 10^9$.

Ví dụ:

NETPATH . INP	NETPATH . OUT
2 2	4
2 4	
6 8	

NETPATH . INP	NETPATH . OUT
2 2	0
2 5	
6 7	

Hạn chế: Trong tất cả các test: $1 < m, n \leq 100$; $a_{ij} \leq 30000$, $i=1,2,\dots,m$; $j=1,2,\dots,n$. Có 50% số lượng test với $m, n \leq 50$.

Bài 3. Mạng máy tính

Một hệ thống n máy tính (các máy tính được đánh số từ 1 đến n) được nối lại thành một mạng bởi m kênh nối, mỗi kênh nối hai máy nào đó và cho phép truyền tin một chiều từ máy này đến máy kia. Giả sử s và t là hai máy tính trong mạng. Ta gọi đường truyền tin từ máy s đến máy t là một dãy các máy tính và các kênh nối chúng có dạng:

$$s = u_1, e_1, u_2, \dots, u_i, e_i, u_{i+1}, \dots, u_{k-1}, e_{k-1}, u_k = t,$$

trong đó u_1, u_2, \dots, u_k là các máy tính trong mạng, e_i – kênh truyền tin từ máy u_i đến máy u_{i+1} ($i = 1, 2, \dots, k-1$).

Mạng máy tính được gọi là **thông suốt** nếu như đối với hai máy u, v bất kỳ ta luôn có đường truyền tin từ u đến v và đường truyền tin từ v đến u . Mạng máy tính được gọi là **hầu như thông suốt** nếu như đối với hai máy u, v bất kỳ, hoặc là có đường truyền tin từ u đến v hoặc là có đường truyền tin từ v đến u .

Biết rằng mạng máy tính đã cho là hầu như thông suốt nhưng không là thông suốt.

Yêu cầu: Hãy xác định xem có thể bổ sung đúng một kênh truyền tin để biến mạng đã cho trở thành thông suốt được hay không?

Dữ liệu: Vào từ file văn bản ONEARC.INP:

- Dòng đầu tiên chứa 2 số nguyên dương n và m .
- Dòng thứ i trong số m dòng tiếp theo mô tả kênh nối thứ i bao gồm hai số nguyên dương u_i, v_i cho biết kênh nối thứ i cho phép truyền tin từ máy u_i đến máy $v_i, i=1,2,\dots,m$.

Các số trên cùng một dòng được ghi cách nhau bởi dấu cách.

Kết quả: Ghi ra file văn bản ONEARC.OUT:

- Dòng đầu tiên ghi ‘YES’ nếu câu trả lời là khẳng định, ghi ‘NO’ nếu câu trả lời là phủ định.
- Nếu câu trả lời là khẳng định thì dòng thứ hai ghi hai số nguyên dương u, v cách nhau bởi dấu cách cho biết cần bổ sung kênh truyền tin từ máy u đến máy v để biến mạng thành thông suốt.

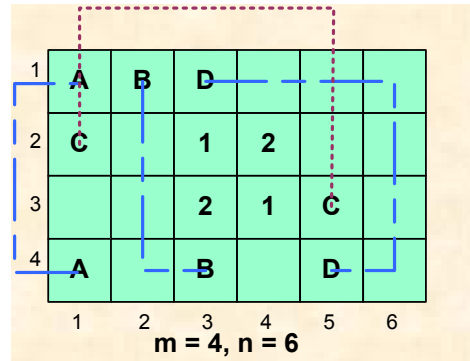
Ví dụ:

ONEARC . INP	ONEARC . OUT
3 2	YES
1 2	3 1
2 3	

Hạn chế: Trong tất cả các test: $n \leq 2000, m \leq 30000$. Có 50% số lượng test với $n \leq 200$.

Bài 4. Xoá cặp ô

Cho một bảng hình chữ nhật kích thước $m \times n$ ô vuông kích thước đơn vị. Các dòng được đánh số từ 1 đến m , từ trên xuống dưới. Các cột được đánh số từ 1 đến n , từ trái qua phải. Ô nằm ở vị trí dòng i và cột j của bảng được gọi là ô (i, j) . Mỗi ô của bảng hoặc để trống hoặc chứa một ký tự lấy từ tập Σ gồm các chữ đến 9 và các chữ cái la tinh in hoa từ A đến Z. Mỗi ký tự xuất hiện ở không quá 4 ô trong bảng. Hai ô chứa một ký tự được gọi là giống nhau. Hai ô giống nhau có xoá được nếu chúng có cạnh chung hoặc tâm (giao của hai đường chéo) của 2 ô này có thể nối với nhau một đường gấp khúc gồm không quá 3 đoạn thẳng độ nguyên, mỗi đoạn song song với cạnh của bảng, và trừ hai ô cần xoá, đường gấp khúc này chỉ qua các ô trống hay nằm ngoài bảng. Các ô bị xoá trở thành ô trống. Mỗi lần xoá một cặp ô của bảng được gọi là một bước. Hình bên nêu ví dụ với trường hợp $m = 4$ và $n = 6$. Bước đầu tiên có thể xoá hai ô chứa ký tự 'A' hoặc 2 ô chứa ký tự 'B' hay 2 ô chứa ký tự 'D'. Hai ô chứa ký tự 'C' chỉ có thể xoá sớm nhất ở bước thứ 2, sau khi đã xoá các ô chứa 'A'. Như vậy, để xoá trống 2 ô $(2, 1)$ và $(1, 2)$ cần thực hiện tối thiểu 3 bước xoá.



được số từ 0 ký tự của cùng thể điểm bằng dài ngoài

Yêu cầu: Cho hai số m, n và m xâu độ dài n mô tả các dòng của bảng và hai ô khác trống $(r_1, c_1), (r_2, c_2)$. Hãy xác định số bước ít nhất cần thực hiện để biến đổi các ô (r_1, c_1) và (r_2, c_2) trở thành ô trống.

Dữ liệu: Vào từ file văn bản DEL.INP:

- Dòng đầu tiên chứa 6 số nguyên m, n, r_1, c_1, r_2, c_2 , hai số liên tiếp được ghi cách nhau bởi dấu cách.
- Dòng thứ $i + 1$ chứa xâu n ký tự mô tả dòng thứ i của bảng ($i = 1, 2, \dots, m$). Các ô trống được thể hiện bằng dấu chấm ('.').

Kết quả: Đưa ra file văn bản DEL.OUT số nguyên k là số bước ít nhất tìm được (qui ước: nếu không tồn tại cách biến đổi thoả mãn yêu cầu đặt ra thì $k=-1$).

Ví dụ:

DEL . INP	DEL . OUT
4 5 2 1 1 2	3
ABD...	
C.12..	
..21C.	
A.B.D.	

DEL . INP	DEL . OUT
4 6 4 2 4 6	3
ABCDUV	
BADCUV	
ABCDUV	
BADCUV	

Hạn chế: Trong tất cả các test: $0 < m \leq 10, 0 < n \leq 20$. Có 60% số lượng test có $m \leq 8, n \leq 10$ và số lượng các ô khác trống không quá $\frac{m \times n}{2}$.

Bảng B

Bài 1. Chọn ô

Cho một bảng hình chữ nhật kích thước $4 \times n$ ô vuông. Các dòng được đánh số từ 1 đến 4, từ trên xuống dưới, các cột được đánh số từ 1 đến n từ trái qua phải. Ô nằm trên giao của dòng i và cột j được gọi là ô (i, j) . Trên mỗi ô (i, j) có ghi một số nguyên a_{ij} , $i = 1, 2, 3, 4; j = 1, 2, \dots, n$. Một cách chọn ô là việc xác định một tập con khác rỗng S của tập tất cả các ô của bảng sao cho không có hai ô nào trong S có chung cạnh. Các ô trong tập S được gọi là ô được chọn, tổng các số trong các ô được chọn được gọi là trọng lượng của cách chọn.

Ví dụ: Xét bảng với $n=3$ trong hình vẽ dưới đây

	1	2	3
1	-1	9	3
2	-4	5	-6
3	7	8	9
4	9	7	2

Cách chọn cần tìm là tập các ô $S = \{(3,1), (1,2), (4,2), (3,3)\}$ với trọng lượng 32.

Yêu cầu: Hãy tìm cách chọn ô với trọng lượng lớn nhất.

Dữ liệu: Vào từ file văn bản SELECT.INP:

- Dòng đầu tiên chứa số nguyên dương n là số cột của bảng.
- Dòng thứ j trong số n dòng tiếp theo chứa 4 số nguyên $a_{1j}, a_{2j}, a_{3j}, a_{4j}$, hai số liên tiếp cách nhau ít nhất một dấu cách, là 4 số trên cột j của bảng.

Kết quả: Ghi ra file văn bản SELECT.OUT trọng lượng của cách chọn tìm được.

Ví dụ:

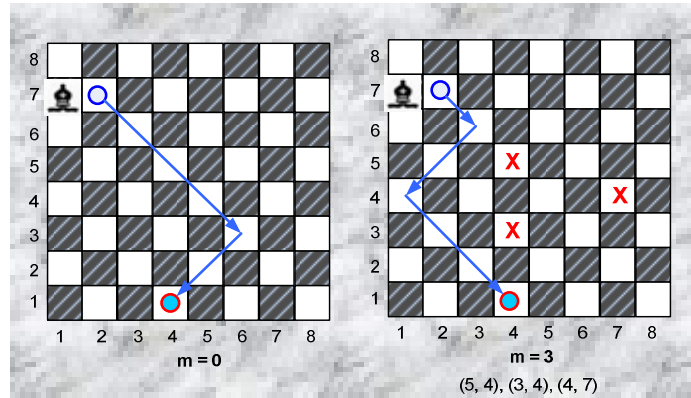
SELECT . INP	SELECT . OUT
3	32
-1 -4 7 9	
9 5 8 7	
3 -6 9 2	

SELECT . INP	SELECT . OUT
3	30
5 5 5 5	
5 5 5 5	
5 5 5 5	

Hạn chế: Trong tất cả các test: $n \leq 10000, |a_{ij}| \leq 30000$. Có 50% số lượng test với $n \leq 1000$.

Bài 2. Quân tượng

Xét bàn cờ vuông kích thước $n \times n$. Các dòng được đánh số từ 1 đến n , từ dưới lên trên. Các cột được đánh số từ 1 đến n từ trái qua phải. Ô nằm trên giao của dòng i và cột j được gọi là ô (i, j) . Trên bàn cờ có m ($0 \leq m \leq n$) quân cờ. Với $m > 0$, quân cờ thứ i ở ô (r_i, c_i) , $i = 1, 2, \dots, m$. Không có hai quân cờ nào ở trên cùng một ô. Trong số các ô còn lại của bàn cờ, tại ô (p, q) có một quân tượng. Mỗi một nước đi, từ vị trí đang đứng quân tượng chỉ có thể di chuyển đến được những ô trên cùng đường chéo với nó mà trên đường đi không phải qua các ô đã có quân.



Cần phải đưa quân tượng từ ô xuất phát (p, q) về ô đích (s, t) . Giả thiết là ở ô đích không có quân cờ. Nếu ngoài quân tượng không có quân nào khác trên bàn cờ thì chỉ có 2 trường hợp: hoặc là không thể tới được ô đích, hoặc là tới được sau không quá 2 nước đi (hình trái). Khi trên bàn cờ còn có các quân cờ khác, vấn đề sẽ không còn đơn giản như vậy.

Yêu cầu: Cho kích thước bàn cờ n , số quân cờ hiện có trên bàn cờ m và vị trí của chúng, ô xuất phát và ô đích của quân tượng. Hãy xác định số nước đi ít nhất cần thực hiện để đưa quân tượng về ô đích hoặc đưa ra số -1 nếu điều này không thể thực hiện được.

Dữ liệu: Vào từ file văn bản BISHOP.INP:

- Dòng đầu tiên chứa 6 số nguyên n, m, p, q, s, t ;
- Nếu $m > 0$ thì mỗi dòng thứ i trong m dòng tiếp theo chứa một cặp số nguyên r_i, c_i xác định vị trí quân thứ i .

Hai số liên tiếp trên cùng một dòng được ghi cách nhau ít nhất một dấu cách.

Kết quả: Đưa ra file văn bản BISHOP.OUT một số nguyên là số nước đi tìm được.

Ví dụ:

BISHOP . INP					
8	3	7	2	1	4
5	4				
3	4				
4	7				

BISHOP . OUT
3

Hạn chế: Trong tất cả các test: $1 \leq n \leq 200$. Có 60% số lượng test với $n \leq 20$.

Bài 3. Kênh xung yếu

Một hệ thống n máy tính (các máy tính được đánh số từ 1 đến n) được nối lại thành một mạng bởi m kênh nối, mỗi kênh nối hai máy nào đó và cho phép truyền tin một chiều từ máy này đến máy kia. Ta gọi một **mạch vòng** của mạng đã cho là một dãy các máy tính và các kênh nối chúng có dạng:

$$u_1, e_1, u_2, \dots, u_i, e_i, u_{i+1}, \dots, u_{k-1}, e_{k-1}, u_k, e_k, u_1$$

trong đó u_1, u_2, \dots, u_k là các máy tính khác nhau trong mạng, e_i – kênh truyền tin từ máy u_i đến máy u_{i+1} ($i = 1, 2, \dots, k-1$), e_k là kênh truyền tin từ máy u_k đến máy u_1 . Một kênh truyền tin trong mạng được gọi là **kênh xung yếu** nếu như bất cứ mạch vòng nào của mạng cũng đều chứa nó.

Yêu cầu: Hãy xác định tất cả các kênh xung yếu của mạng đã cho.

Dữ liệu: Vào từ file văn bản CIRARC.INP:

- Dòng đầu tiên chứa 2 số nguyên dương n và m .
- Dòng thứ i trong số m dòng tiếp theo mô tả kênh nối thứ i bao gồm hai số nguyên dương u_i, v_i cho biết kênh nối thứ i cho phép truyền tin từ máy u_i đến máy v_i .

Các số trên cùng một dòng được ghi cách nhau bởi dấu cách.

Kết quả: Ghi ra file văn bản CIRARC.OUT:

- Dòng đầu tiên ghi số nguyên k là số lượng kênh xung yếu trong mạng đã cho. Ghi $k = -1$ nếu mạng không chứa kênh xung yếu.
- Nếu $k > 0$ thì mỗi dòng trong số k dòng tiếp theo ghi thông tin về một kênh xung yếu tìm được theo qui cách mô tả giống như trong file dữ liệu vào.

Ví dụ:

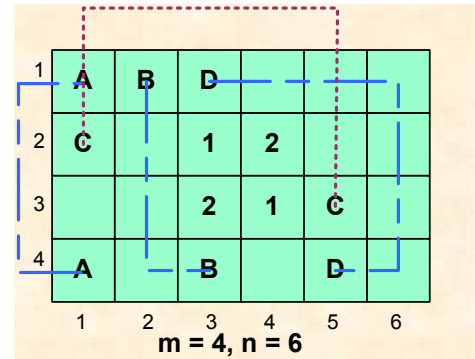
CIRARC . INP	CIRARC . OUT
2 2	2
1 2	1 2
2 1	2 1

CIRARC . INP	CIRARC . OUT
3 3	-1
1 2	
2 3	
1 3	

Hạn chế: Trong tất cả các test: $n \leq 1000, m \leq 20000$. Có 50% số lượng test với $n \leq 200$.

Bài 4. Biến đổi bảng

Cho một bảng hình chữ nhật kích thước $m \times n$ ô vuông kích thước đơn vị. Các dòng được đánh số từ 1 đến m , từ trên xuống dưới. Các cột được đánh số từ 1 đến n , từ trái qua phải. Mỗi ô của bảng hoặc được để trống hoặc chứa một ký tự lấy từ tập Σ gồm các số từ 0 đến 9 và các chữ cái la tinh in hoa từ A đến Z. Hai ô giống nhau có thể xoá được nếu chúng có cạnh chung hoặc có nối các tâm (giao điểm của hai đường chéo) của chúng nhau bằng một đường gấp khúc gồm không quá 3 đoạn thẳng độ dài nguyên, mỗi đoạn song song với cạnh của bảng và ngoại trừ hai ô cần xoá, đường gấp khúc chỉ qua ô trống hay nằm ngoài bảng. Các ô bị xoá trở thành ô trống. Mỗi lần xoá một cặp ô của bảng được gọi là một bước. Hình bên nêu ví dụ với trường hợp $m = 4$ và $n = 6$. Bước đầu tiên có thể xoá hai ô chứa ký tự 'A', tiếp theo, lần lượt xoá các cặp ô chứa 'B', chứa 'C' và cặp ô chứa 'D'. Ở ví dụ này, sau khi thực hiện 4 bước xoá có thể, trong bảng còn lại 4 ô không thể xoá được.



chữ
ô
của
thể
với
các

Yêu cầu: Cho m, n và m xâu độ dài n mô tả các dòng của bảng. Hãy xác định số lượng ô lớn nhất có thể xoá được.

Dữ liệu: Vào từ file văn bản CHANGE.INP:

- Dòng đầu tiên chứa 2 số nguyên m, n được ghi cách nhau bởi dấu cách.
- Dòng thứ $i+1$ chứa xâu n ký tự mô tả dòng thứ i của bảng ($i = 1, 2, \dots, m$). Các ô trống được thể hiện bằng dấu chấm ('.').

Kết quả: Đưa ra file văn bản CHANGE.OUT một số nguyên là số lượng ô lớn nhất có thể xoá được.

Ví dụ:

CHANGE . INP	CHANGE . OUT
4 5	8
ABD...	
C.12..	
..21C.	
A.B.D.	

CHANGE . INP	CHANGE . OUT
4 6	24
ABCDUV	
BADCUV	
ABCDUV	
BADCUV	

Hạn chế: Trong tất cả các test: $0 < m \leq 10, 0 < n \leq 10$. Có 60% số lượng test có $m \leq 5, n \leq 6$ và số lượng các ô khác trống không quá $\frac{m \times n}{2}$.

Năm 2007

Bài 1. Dãy con không giảm dài nhất (6 điểm)

Cho dãy số nguyên dương a_1, a_2, \dots, a_n .

Dãy số

$$a_i, a_{i+1}, \dots, a_j \text{ thỏa mãn } a_i \leq a_{i+1} \leq \dots \leq a_j,$$

với $1 \leq i \leq j \leq n$ được gọi là *dãy con không giảm* của dãy số đã cho và khi đó số $j-i+1$ được gọi là *độ dài* của dãy con này.

Yêu cầu: Trong số các dãy con không giảm của dãy số đã cho mà các phần tử của nó đều thuộc dãy số $\{u_k\}$ xác định bởi $u_1 = 1, u_k = u_{k-1} + k (k \geq 2)$, hãy tìm dãy con có độ dài lớn nhất.

Dữ liệu: Vào từ file văn bản MAXISEQ.INP

- Dòng đầu tiên chứa một số nguyên dương $n (n \leq 10^4)$;
- Dòng thứ i trong n dòng tiếp theo chứa một số nguyên dương $a_i (a_i \leq 10^8)$ là số hạng thứ i của dãy số đã cho, $i = 1, 2, \dots, n$.

Kết quả: Ghi ra file văn bản MAXISEQ.OUT số nguyên d là độ dài của dãy con không giảm tìm được (quy ước rằng nếu không có dãy con nào thỏa mãn điều kiện đặt ra thì $d = 0$).

Ví dụ: Cho dãy số có 8 phần tử: 2, 2007, 6, 6, 15, 16, 3, 21. Các dãy con không giảm của dãy số đã cho mà các phần tử của nó đều thuộc dãy $\{u_k\}$ là: 6, 6, 15 và 3, 21 (vì $u_2 = 3, u_3 = 6, u_5 = 15, u_6 = 21$). Dãy cần tìm là 6, 6, 15 có độ dài là 3.

MAXISEQ.INP	MAXISEQ.OUT
8	3
2	
2007	
6	
6	
15	
16	
3	
21	

Bài 2. Siêu thị may mắn (7 điểm)

An được mời tham gia trò chơi “Siêu thị may mắn” do đài truyền hình ZTV tổ chức. Siêu thị được đặt trong trường quay truyền hình có n mặt hàng được đánh số từ 1 đến n và mặt hàng thứ i được niêm yết giá là c_i đồng, $i = 1, 2, \dots, n$. Theo thể lệ của trò chơi, An được ban tổ chức tặng một thẻ mua hàng có giá trị là s đồng và phải dùng hết số tiền trong thẻ này để mua hàng trong siêu thị với điều kiện mặt hàng thứ i chỉ được mua với số lượng nhiều nhất là m_i , $i = 1, 2, \dots, n$. An sẽ là người thắng cuộc nếu tìm được tổng số cách mua hàng thỏa mãn yêu cầu đặt ra và chỉ ra một cách mua hàng nếu có.

Yêu cầu: Hãy giúp An trở thành người thắng cuộc khi cho bạn biết trước các giá trị n , s , c_i và m_i ($1 \leq n \leq 500$; $1 \leq s \leq 10^5$; $1 \leq c_i \leq 10^4$; $1 \leq m_i \leq 100$) với $i = 1, 2, \dots, n$.

Dữ liệu: Vào từ file văn bản SMARKET.INP

- Dòng đầu tiên chứa hai số nguyên dương s và n ;
- Dòng thứ i trong n dòng tiếp theo chứa hai số nguyên dương c_i và m_i với $i = 1, 2, \dots, n$.

Kết quả: Ghi ra file văn bản SMARKET.OUT

- Dòng đầu tiên ghi số nguyên d là tổng số cách mua hàng tìm được;
- Nếu $d \geq 1$ thì dòng thứ hai ghi một cách mua hàng tìm được là một dãy n số nguyên, trong đó số hạng thứ i là số lượng mặt hàng thứ i mua được trong cách mua hàng này, $i = 1, 2, \dots, n$.

Hai số liên tiếp trên một dòng trong file dữ liệu và file kết quả cách nhau ít nhất một dấu cách.

Ví dụ:

SMARKET . INP	SMARKET . OUT
12 3	2
4 1	0 2 0
6 2	
2 1	

Bài 3. Robot cứu hoả (7 điểm)

Trên một mạng lưới giao thông có n nút, các nút được đánh số từ 1 đến n và giữa hai nút bất kỳ có không quá một đường nối trực tiếp (đường nối trực tiếp là một đường hai chiều). Ta gọi đường đi từ nút s đến nút t là một dãy các nút và các đường nối trực tiếp có dạng:

$$s = u_1, e_1, u_2, \dots, u_i, e_i, u_{i+1}, \dots, u_{k-1}, e_{k-1}, u_k = t,$$

trong đó u_1, u_2, \dots, u_k là các nút trong mạng lưới giao thông, e_i là đường nối trực tiếp giữa nút u_i và u_{i+1} (không có nút u_j nào xuất hiện nhiều hơn một lần trong dãy trên, $j = 1, 2, \dots, k$).

Biết rằng mạng lưới giao thông được xét luôn có ít nhất một đường đi từ nút 1 đến nút n .

Một robot chứa đầy bình với w đơn vị năng lượng, cần đi từ trạm cứu hoả đặt tại nút 1 đến nơi xảy ra hoả hoạn ở nút n , trong thời gian ít nhất có thể. Thời gian và chi phí năng lượng để robot đi trên đường nối trực tiếp từ nút i đến nút j tương ứng là t_{ij} và c_{ij} ($1 \leq i, j \leq n$). Robot chỉ có thể đi được trên đường nối trực tiếp từ nút i đến nút j nếu năng lượng còn lại trong bình chứa không ít hơn c_{ij} ($1 \leq i, j \leq n$). Nếu robot đi đến một nút có trạm tiếp năng lượng (một nút có thể có hoặc không có trạm tiếp năng lượng) thì nó tự động được nạp đầy năng lượng vào bình chứa với thời gian nạp coi như không đáng kể.

Yêu cầu: Hãy xác định giá trị w nhỏ nhất để robot đi được trên một đường đi từ nút 1 đến nút n trong thời gian ít nhất.

Dữ liệu: Vào từ file văn bản ROBOT.INP

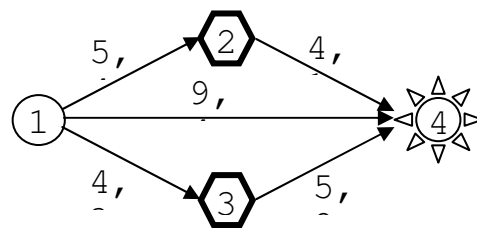
- Dòng đầu tiên chứa một số nguyên dương n ($2 \leq n \leq 500$);
- Dòng thứ hai chứa n số, trong đó số thứ j bằng 1 hoặc 0 tương ứng ở nút j có hoặc không có trạm tiếp năng lượng ($j = 1, 2, \dots, n$);
- Dòng thứ ba chứa số nguyên dương m ($m \leq 30000$) là số đường nối trực tiếp có trong mạng lưới giao thông;
- Dòng thứ k trong số m dòng tiếp theo chứa 4 số nguyên dương i, j, t_{ij}, c_{ij} ($t_{ij}, c_{ij} \leq 10^4$) mô tả đường nối trực tiếp từ nút i đến nút j , thời gian và chi phí năng lượng tương ứng.

Hai số liên tiếp trên một dòng trong file dữ liệu cách nhau ít nhất một dấu cách.

Kết quả: Ghi ra file văn bản ROBOT.OUT một số nguyên dương w tìm được.

Ví dụ:

ROBOT . INP	ROBOT . OUT
4	3
0 1 1 0	
5	
1 2 5 4	
1 3 4 3	
1 4 9 4	
2 4 4 1	
3 4 5 2	



Nút 2 và nút 3 có trạm tiếp năng lượng

Năm 2008

Bài 1. Trò chơi với dãy số (6 điểm)

Hai bạn học sinh trong lúc nhàn rỗi nghĩ ra trò chơi sau đây. Mỗi bạn chọn trước một dãy số gồm n số nguyên. Giả sử dãy số mà bạn thứ nhất chọn là

$$b_1, b_2, \dots, b_n$$

còn dãy số mà bạn thứ hai chọn là

$$c_1, c_2, \dots, c_n.$$

Mỗi lượt chơi mỗi bạn đưa ra một số hạng trong dãy số của mình. Nếu bạn thứ nhất đưa ra số hạng b_i ($1 \leq i \leq n$), còn bạn thứ hai đưa ra số hạng c_j ($1 \leq j \leq n$) thì giá của lượt chơi đó sẽ là $|b_i + c_j|$.

Ví dụ: Giả sử dãy số bạn thứ nhất chọn là 1, - 2; còn dãy số mà bạn thứ hai chọn là 2, 3. Khi đó các khả năng có thể của một lượt chơi là (1, 2), (1, 3), (-2, 2), (-2, 3). Như vậy, giá nhỏ nhất của một lượt chơi trong số các lượt chơi có thể là 0 tương ứng với giá của lượt chơi (-2, 2).

Yêu cầu: Hãy xác định giá nhỏ nhất của một lượt chơi trong số các lượt chơi có thể.

Dữ liệu: Vào từ file văn bản SEQGAME.INP:

- Dòng đầu tiên chứa số nguyên dương n ($n \leq 10^5$);
- Dòng thứ hai chứa dãy số nguyên b_1, b_2, \dots, b_n ($|b_i| \leq 10^9, i = 1, 2, \dots, n$);
- Dòng thứ ba chứa dãy số nguyên c_1, c_2, \dots, c_n ($|c_j| \leq 10^9, j = 1, 2, \dots, n$).

Hai số liên tiếp trên một dòng được ghi cách nhau bởi dấu cách.

Kết quả: Ghi ra file văn bản SEQGAME.OUT giá nhỏ nhất tìm được.

Ví dụ:

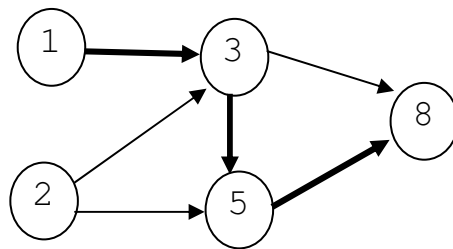
SEQGAME . INP	SEQGAME . OUT
2	0
1 -2	
2 3	

Ràng buộc: 60% số tests ứng với 60% số điểm của bài có $1 \leq n \leq 1000$.

Bài 2. Lò cò (7 điểm)

Nhảy lò cò là trò chơi dân gian của Việt Nam. Người trên hành tinh X cũng rất thích trò chơi này và họ đã cải biên trò chơi này như sau: Trên mặt phẳng vẽ n vòng tròn được đánh số từ 1 đến n . Tại vòng tròn i người ta điền số nguyên dương a_i . Hai số trên hai vòng tròn tùy ý không nhất thiết phải khác nhau. Tiếp đến người ta vẽ các mũi tên, mỗi mũi tên hướng từ một vòng tròn đến một vòng tròn khác. Quy tắc vẽ mũi tên là: Nếu có ba số a_i, a_j, a_k thỏa mãn: $a_k = a_i + a_j$ thì vẽ mũi tên hướng từ vòng tròn i đến vòng tròn k và mũi tên hướng từ vòng tròn j đến vòng tròn k . Người chơi chỉ được di chuyển từ một vòng tròn đến vòng tròn khác nếu có mũi tên từ vòng tròn xuất phát hướng đến vòng tròn đích. Người chơi có thể xuất phát từ một trong số các vòng tròn, di chuyển theo các mũi tên đã vẽ để đến các vòng tròn khác. Người thắng cuộc sẽ là người tìm được cách di chuyển qua nhiều vòng tròn nhất.

Ví dụ: Với 5 vòng tròn và các số trong vòng tròn là 1, 2, 8, 3, 5, trò chơi được trình bày trong hình dưới đây:



Khi đó có thể di chuyển được nhiều nhất qua 4 vòng tròn (tương ứng với đường di chuyển được tô đậm trên hình vẽ).

Yêu cầu: Hãy xác định xem trong trò chơi mô tả ở trên, nhiều nhất có thể di chuyển được qua bao nhiêu vòng tròn.

Dữ liệu: Vào từ file văn bản JUMP.INP:

- Dòng đầu tiên chứa số nguyên n ($3 \leq n \leq 1000$);
- Dòng thứ hai chứa dãy số nguyên dương a_1, a_2, \dots, a_n ($a_i \leq 10^9, i = 1, 2, \dots, n$).

Hai số liên tiếp trên một dòng được ghi cách nhau bởi dấu cách.

Kết quả: Ghi ra file văn bản JUMP.OUT số lượng vòng tròn trên đường di chuyển tìm được.

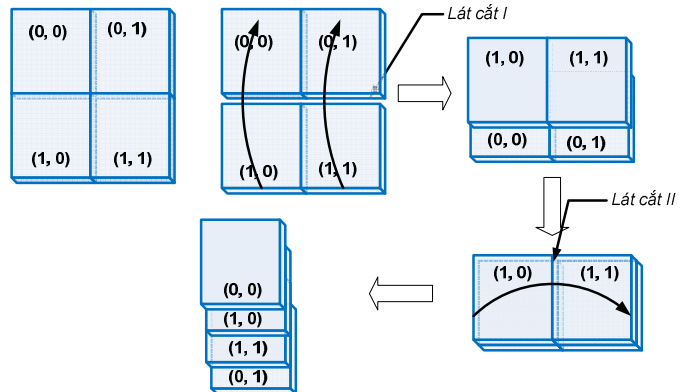
Ví dụ:

JUMP . INP	JUMP . OUT
5	4
1 2 8 3 5	

Ràng buộc: 60% số tests ứng với 60% số điểm của bài có $3 \leq n \leq 100$.

Bài 3. Quà Tết (7 điểm)

Chuẩn bị đón năm mới, Công ty bánh kẹo Hương Dừa đã làm một tấm sôcôla cực lớn với mục đích ghi tên mình vào sách kỷ lục Ghi nét đồng thời quảng bá thương hiệu trước công chúng. Tấm sôcôla có hình vuông kích thước $2^k \times 2^k$ ô, tạo thành lưới ô vuông 2^k hàng và 2^k cột. Các hàng được đánh số từ 0 đến $2^k - 1$ từ trên xuống dưới, các cột được đánh số từ 0 đến $2^k - 1$ từ trái sang phải. Ô nằm trên hàng i và cột j được gọi là ô (i, j) . Sau buổi trưng bày giới thiệu sản phẩm, tấm sôcôla được cắt nhỏ, chia cho mọi người, mỗi người được một ô của chiếc bánh kỷ lục. Bộ phận tiếp thị đã ấn vào hai ô khác nhau (p, q) và (u, v) mỗi ô một đồng xu. Vị khách nào may mắn nhận được ô sôcôla có đồng xu sẽ được tặng rất nhiều sản phẩm độc đáo của Công ty.



Vì chiếc bánh rất lớn nên Công ty đã thiết kế một máy cắt bánh. Máy thực hiện dãy các thao tác cắt, bắt đầu từ chồng bánh chỉ gồm 1 tấm sôcôla ban đầu, mỗi thao tác gồm hai bước sau:

- Bước 1: Cắt ngang song song với cạnh chồng bánh chia chồng sôcôla thành hai phần bằng nhau, úp chồng bánh bên dưới lên chồng bánh bên trên sao cho mép dưới đè lên mép trên.
- Bước 2: Cắt dọc song song với cạnh chồng bánh chia chồng sôcôla thành hai phần bằng nhau, úp chồng bánh bên trái lên trên chồng bánh bên phải sao cho mép trái đè lên mép phải.

Như vậy sau mỗi lần thực hiện thao tác cắt, chiều dài và chiều rộng của các tấm sôcôla giảm đi một nửa. Sau k lần thực hiện thao tác cắt, các ô của tấm sôcôla sẽ được xếp thành một cột. Khách nhận bánh xếp hàng một và được đánh số từ 1 trở đi, người thứ m sẽ nhận được ô sôcôla thứ m từ trên xuống dưới ($1 \leq m \leq 2^k \times 2^k$).

Ví dụ, với $k = 1$ và đồng xu được ấn vào các ô $(0, 0)$, $(1, 1)$, việc thực hiện các thao tác cắt được trình bày trên hình vẽ minh họa ở trên. Trong ví dụ này, vị khách thứ nhất và thứ ba sẽ là những người nhận được tặng phẩm của Công ty.

Yêu cầu: Cho biết các số nguyên k, p, q, u, v . Hãy xác định số thứ tự của hai vị khách may mắn được nhận quà.

Dữ liệu: Vào từ file văn bản GIFTS.INP gồm một dòng chứa 5 số nguyên k, p, q, u, v , các số cách nhau bởi dấu cách.

Kết quả: Đưa ra file văn bản GIFTS.OUT một dòng chứa hai số nguyên là số thứ tự của các vị khách may mắn. Hai số phải cách nhau đúng một dấu cách.

Ví dụ:

GIFTS . INP	GIFTS . OUT
1 0 0 1 1	1 3

Ràng buộc:

- $1 \leq k \leq 40, 0 \leq p, q, u, v \leq 2^k - 1$;
- 60% số tests ứng với 60% số điểm của bài có $1 \leq k \leq 5$.

Đề thi vòng II quốc gia

Năm 2005

Bài 1. Lập lịch

Tại thời điểm 0, cần phải sắp xếp thực hiện n chương trình (đánh số từ 1 đến n) trên máy tính. Chương trình i có thời hạn hoàn thành là d_i và đòi hỏi thời gian thực hiện là p_i (d_i và p_i là các số nguyên dương). Việc thực hiện chương trình trên máy tính cần được tiến hành liên tục từ thời điểm bắt đầu đến đến thời điểm kết thúc. Khoảng thời gian thực hiện hai chương trình bất kỳ không được có nhiều hơn một điểm chung (nghĩa là hai khoảng không được giao nhau, ngoại trừ tình huống thời điểm cuối của khoảng này là trùng với thời điểm đầu của khoảng kia). Giả sử chương trình i bắt đầu thực hiện từ thời điểm s_i , khi đó $f_i = s_i + p_i$ được gọi là thời điểm hoàn thành của nó. Ta nói chương trình i là hoàn thành đúng hạn, nếu thời điểm hoàn thành của nó là không lớn hơn d_i .

Yêu cầu: Hãy tìm trình tự thực hiện các chương trình sao cho có nhiều chương trình được hoàn thành đúng hạn nhất.

Dữ liệu: Vào từ file văn bản SCHEDULE.INP:

- Dòng đầu tiên chứa số nguyên dương n ($n \leq 1000$);
- Dòng thứ i trong số n dòng tiếp theo chứa thông tin về chương trình thứ i gồm hai số p_i, d_i được ghi cách nhau bởi dấu cách ($p_i \leq d_i \leq 20000$).

Các số trên cùng dòng được ghi cách nhau bởi ít nhất một dấu cách.

Kết quả: Ghi ra file văn bản SCHEDULE.OUT số lượng chương trình được hoàn thành đúng hạn theo trình tự tìm được.

Ví dụ:

SCHEDULE . INP
3
3 3
2 4
2 4

SCHEDULE . OUT
2

Bài 2. Mạng đen trắng

Một hệ thống gồm n máy tính (được đánh số từ 1 đến n) được nối với nhau thành mạng bởi các kênh nối (cho phép truyền tin hai chiều) giữa một số cặp máy tính. Mạng đã cho là *thông suốt*, nghĩa là các kênh nối đảm bảo hai máy tính bất kỳ có thể truyền tin với nhau qua kênh nối trực tiếp giữa chúng hoặc thông qua một số máy trung gian. Các kênh nối giữa các cặp máy chỉ thuộc một trong hai loại trắng hoặc đen mà để cho tiện ta sẽ gọi là kênh trắng hoặc kênh đen. Ta gọi một **phép rút gọn** mạng đã cho là một cách loại bỏ khỏi mạng một số nhiều nhất các kênh nối sao cho mạng còn lại vẫn là thông suốt.

Yêu cầu: Trong số các phép rút gọn mạng đã cho hãy tìm cách rút gọn sao cho mạng còn lại có chênh lệch giữa số lượng kênh trắng và số lượng kênh đen (trị tuyệt đối của hiệu giữa số lượng kênh trắng và kênh đen) là nhỏ nhất. (Nếu có nhiều lời giải thì chỉ cần đưa ra một lời giải tùy ý.)

Dữ liệu: Vào từ file văn bản BWNET.INP:

- Dòng đầu tiên chứa 2 số nguyên dương n và m , trong đó n là số lượng máy tính còn m là số lượng kênh nối trong mạng đã cho ($2 < n \leq 200$).
- Mỗi dòng trong số m dòng tiếp theo chứa thông tin về một kênh nối bao gồm 3 số nguyên dương u, v, k , cho biết hai máy u, v được nối với nhau bởi kênh nối loại k ($k=1$ nếu là kênh đen và $k=2$ nếu là kênh trắng).

Các số trên cùng dòng được ghi cách nhau bởi ít nhất một dấu cách.

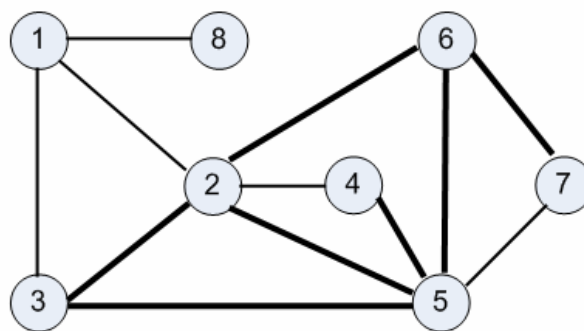
Kết quả: Ghi ra file văn bản BWNET.OUT:

- Dòng đầu tiên ghi hai số nguyên p, q , trong đó p là số kênh nối còn q là chênh lệch giữa số lượng kênh trắng và kênh đen trong mạng tìm được.
- p dòng tiếp theo mô tả p kênh nối của mạng tìm được theo qui cách giống như trong file dữ liệu.

Ví dụ:

BWNET . INP	BWNET . OUT
8 12	7 1
1 8 1	3 1 1
1 2 1	4 5 2
1 3 1	5 3 2
2 3 2	6 5 2
2 4 1	7 6 2
2 5 2	2 1 1
2 6 2	8 1 1
3 5 2	
4 5 2	
5 6 2	
5 7 1	
6 7 2	

Sơ đồ nối mạng



Bài 3. Biến đổi

Cho xâu ký tự S độ dài L ($L \leq 20000$) chỉ bao gồm các chữ cái la tinh in hoa và các chữ số thập phân. Ta gọi *đoạn chữ cái* là dãy liên tiếp các ký tự chữ cái $S_M, S_{M+1}, \dots, S_{M+P-1}$ trong xâu S thoả mãn điều kiện: ký tự đứng trước S_M (nếu có) và ký tự đứng sau S_{M+P-1} (nếu có) phải là chữ số. Tương tự như vậy, *đoạn chữ số* là dãy liên tiếp các ký tự chữ số $S_N, S_{N+1}, \dots, S_{N+Q-1}$ trong xâu S thoả mãn điều kiện: ký tự đứng trước S_N (nếu có) và ký tự đứng sau S_{N+Q-1} (nếu có) phải là chữ cái.

Có 6 phép biến đổi tác động lên S (như là ví dụ lấy $S = \text{'ABC1234EF5'}$):

1. Dịch chuyển vòng tròn các ký tự của S theo chiều kim đồng hồ: Ký tự S_L trở thành ký tự đầu của xâu: $\text{ABC1234EF5} \rightarrow \text{5ABC1234EF}$.
2. Dịch chuyển vòng tròn các ký tự của S theo chiều ngược kim đồng hồ: Ký tự S_1 trở thành ký tự cuối xâu: $\text{ABC1234EF5} \rightarrow \text{BC1234EF5A}$,
3. Trong một đoạn chữ số tùy chọn thay mỗi ký tự số bằng ký tự số kế tiếp theo qui tắc: Ký tự 0 - bằng ký tự 1, Ký tự 1 - bằng ký tự 2, . . . , Ký tự 9 - bằng ký tự 0,
4. Trong một đoạn chữ số tùy chọn thay mỗi ký tự số bằng ký tự số đi trước theo qui tắc: Ký tự 1 - bằng ký tự 0, Ký tự 2 - bằng ký tự 1, . . . , Ký tự 0 - bằng ký tự 9,
5. Trong một đoạn chữ cái tùy chọn thay mỗi ký tự chữ cái bằng ký tự chữ cái đi sau theo qui tắc: Ký tự A - bằng ký tự B, Ký tự B - bằng ký tự C, . . . , Ký tự Z - bằng ký tự A,
6. Trong một đoạn chữ cái tùy chọn thay mỗi ký tự chữ cái bằng ký tự chữ cái đi trước theo qui tắc: Ký tự B - bằng ký tự A, Ký tự C - bằng ký tự B, . . . , Ký tự A - bằng ký tự Z.

Yêu cầu: Cho hai xâu $S1$ và $S2$ cùng độ dài L , trong đó $S2$ là kết quả biến đổi $S1$ bằng cách áp dụng một số lần các phép biến đổi nêu trên. Hãy xác định số lượng tối thiểu lần biến đổi để nhận được $S2$ từ $S1$.

Dữ liệu: Vào từ file văn bản CHANGE.INP gồm 2 dòng: dòng thứ nhất chứa xâu $S1$, dòng thứ 2 chứa xâu $S2$.

Kết quả: Đưa ra file văn bản CHANGE.OUT số lượng tối thiểu lần biến đổi để nhận được $S2$ từ $S1$

Ví dụ:

CHANGE . INP
ABC1234EF5
BC1234EF6A

CHANGE . OUT
2

Bài 4. Hái nấm

Một cô bé đi hái nấm trong N khu rừng đánh số từ 1 đến N . Khu rừng thứ i có S_i cây nấm. Việc di chuyển từ khu rừng thứ i sang khu rừng thứ j tốn t_{ij} đơn vị thời gian. Đến mỗi khu rừng, cô bé có thể dừng lại để hái nấm. Nếu tổng số đơn vị thời gian cô bé dừng lại ở khu rừng thứ i là d_i ($d_i > 0$), thì cô bé hái được:

$$\left\lfloor \frac{S_i}{2} \right\rfloor + \left\lfloor \frac{S_i}{4} \right\rfloor + \dots + \left\lfloor \frac{S_i}{2^{d_i}} \right\rfloor$$

cây nấm tại khu rừng đó ($\lfloor x \rfloor$ là phần nguyên của x). Giả thiết rằng ban đầu cô bé ở khu rừng thứ nhất và đi hái nấm trong thời gian không quá P đơn vị.

Yêu cầu: Hãy tính số lượng cây nấm nhiều nhất mà cô bé có thể hái được.

Dữ liệu: Vào từ file văn bản HAINAM.INP:

- Dòng đầu tiên chứa số nguyên dương N ($N \leq 100$) và P ($P \leq 1000$);
- Dòng tiếp theo chứa N số nguyên dương S_i ($S_i \leq 10000$);
- Dòng thứ i trong N dòng cuối cùng chứa N số nguyên dương t_{ij} ($t_{ij} \leq 1000$), ($i, j = 1, \dots, N$).

Các số trên cùng dòng được ghi cách nhau bởi ít nhất một dấu cách.

Kết quả: Ghi ra file văn bản HAINAM.OUT số lượng cây nấm nhiều nhất cô bé có thể hái được.

Ví dụ:

HAINAM . INP	
2	2
5	1000
0	3
3	0

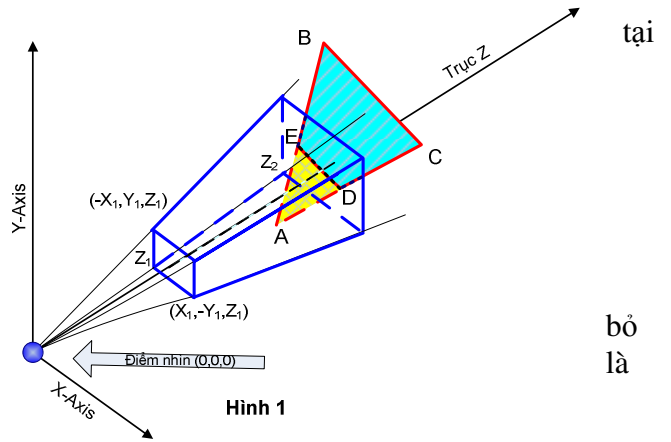
HAINAM . OUT	
3	

Bài 5. Đồ họa 3D

Trong lĩnh vực đồ họa máy tính, các hình khối được biểu diễn qua các đa giác. Trước khi hiển thị các đa giác ra card màn hình, cần phải thực hiện một số bước chuẩn bị. Hai trong những bước chuẩn bị quan trọng là: loại bỏ phần nằm ngoài vùng nhìn và tìm ra thứ tự của các hình khối theo vị trí tương đối với điểm nhìn.

“Vùng nhìn” là một hình chóp cụt có đỉnh $(0,0,0)$; đáy trên là hình chữ nhật trên mặt phẳng $z=Z_1$, với đỉnh trái trên tại $(-X_1,Y_1,Z_1)$, đỉnh phải dưới tại $(X_1,-Y_1,Z_1)$; đáy dưới là hình chữ nhật trên mặt phẳng $z=Z_2$.

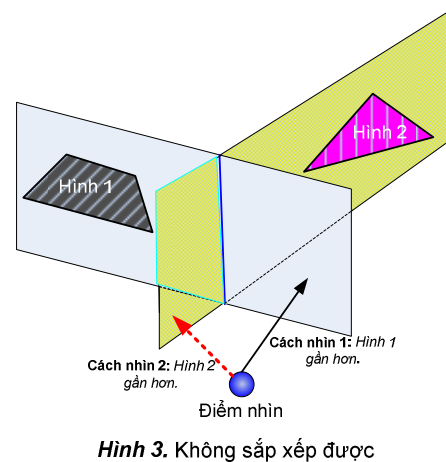
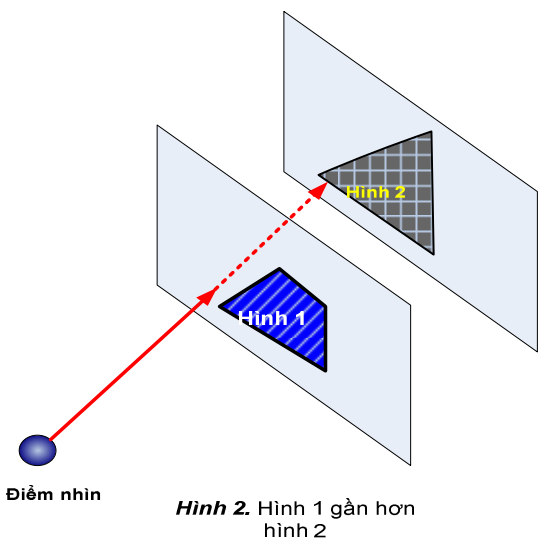
Với mỗi đa giác cần vẽ, các phần của đa giác nằm ngoài “vùng nhìn” sẽ bị loại bỏ để tạo thành một đa giác mới nằm hoàn toàn trong “vùng nhìn”. Ví dụ ở hình 1, tam giác ABC sau khi loại phần BCDE nằm ngoài “vùng nhìn”, phần còn lại tam giác AED.



Sau khi các đa giác đã được cắt bỏ phần nằm ngoài “vùng nhìn”, chúng phải được sắp xếp lại theo vị trí tương đối với điểm nhìn. Điểm nhìn được đặt tại vị trí $(0,0,0)$. Để xác định thứ tự (gần, xa) của hai đa giác $ĐG_1$ và $ĐG_2$, người ta kiểm tra các điều kiện:

- **ĐK1:** điểm nhìn và đa giác $ĐG_2$ nằm ở hai nửa không gian khác nhau ngăn cách bởi mặt phẳng chứa đa giác $ĐG_1$ và không có điểm chung với mặt phẳng chứa $ĐG_1$; hoặc điểm nhìn và đa giác $ĐG_1$ nằm ở cùng nửa không gian xác định bởi mặt phẳng chứa đa giác $ĐG_2$ và không có điểm chung với mặt phẳng chứa $ĐG_2$.
- **ĐK2:** điểm nhìn và đa giác $ĐG_1$ nằm ở hai nửa không gian khác nhau ngăn cách bởi mặt phẳng chứa đa giác $ĐG_2$ và không có điểm chung với mặt phẳng chứa $ĐG_2$; hoặc điểm nhìn và đa giác $ĐG_2$ nằm ở cùng nửa không gian xác định bởi mặt phẳng chứa đa giác $ĐG_1$ và không có điểm chung với mặt phẳng chứa $ĐG_1$.

Nếu chỉ **ĐK1** đúng (đa giác $ĐG_1$ nằm gần điểm nhìn hơn đa giác $ĐG_2$), hoặc chỉ **ĐK2** đúng (đa giác $ĐG_2$ nằm gần điểm nhìn hơn đa giác $ĐG_1$) thì thứ tự của hai đa giác $ĐG_1$ và $ĐG_2$ có thể xác định được (xem ví dụ ở hình 2). Nếu cả **ĐK1** và **ĐK2** cùng đúng hoặc cùng sai, thứ tự của hai đa giác $ĐG_1$ và $ĐG_2$ không thể xác định được (xem ví dụ ở hình 3).



Yêu cầu: Cho N hình tam giác đánh số từ 1 đến N . Hãy loại bỏ các phần nằm ngoài “vùng nhìn” của tam giác này và sau đó sắp xếp lại theo thứ tự ở từ gần đến xa đối với điểm nhìn.

Dữ liệu: Vào từ file văn bản DOHOA.INP:

- Dòng đầu tiên chứa số nguyên dương N ($N \leq 100$);
- Dòng tiếp theo chứa 4 số nguyên dương X_1, Y_1, Z_1, Z_2 ($X_1, Y_1, Z_1, Z_2 \leq 10000$);

- Dòng thứ i trong số N dòng tiếp theo chứa chín số nguyên là tọa độ của ba đỉnh của tam giác thứ i ($i=1, \dots, N$). Các tọa độ có trị tuyệt đối không vượt quá 10000.

Các số trên cùng dòng được ghi cách nhau bởi ít nhất một dấu cách.

Kết quả: Ghi ra file văn bản DOHOA.OUT:

- Nếu có thể xác định được thứ tự giữa mọi cặp hai đa giác được tạo ra từ các tam giác ban đầu sau khi loại bỏ phần nằm ngoài vùng nhìn và có thể sắp xếp được các đa giác này theo định nghĩa thứ tự đó, hãy ghi ra chỉ số của các đa giác, theo thứ tự từ gần đến xa đối với điểm nhìn.
- Nếu không, chỉ ghi ra số -1.

Các số được ghi trên cùng một dòng, cách nhau bởi dấu cách.

Ví dụ:

DOHOA . INP								
2								
1	1	1	4					
1	1	2	1	3	2	4	4	2
1	1	3	1	3	3	3	3	3

DOHOA . OUT	
1	2

Ghi chú: Một dạng phương trình mặt phẳng: $Ax+By+Cz+D=0$. Một dạng phương trình đường thẳng: $x=A_1t+A_2, y=B_1t+B_2, z=C_1t+C_2$ trong đó t là một tham số. (x,y,z) là tọa độ các điểm trên mặt phẳng/đường thẳng đó, $A, B, C, D, A_1, A_2, B_1, B_2, C_1, C_2$ là các hằng số.

Bài 6. Chia nhóm

Các trường tổ chức cho N học sinh đi cắm trại. Cần chia các học sinh này thành các nhóm không nhất thiết phải có số lượng như nhau, mỗi nhóm một trại. Mỗi nhóm phải có ít nhất một người. Số lượng nhóm và số học sinh trong mỗi nhóm sẽ được thông báo cho nhà tài trợ. Dựa trên số liệu này, nhà tài trợ sẽ tính toán số lượng *tối thiểu* chai nước uống để cấp phát sao cho: Mỗi trại đều nhận được số lượng chai nước như nhau; Số lượng các chai nước đó có thể chia đều cho các thành viên của mỗi trại. Các trường muốn tìm cách chia nhóm sao cho mỗi nhóm nhận được nhiều chai nước nhất.

Ví dụ: Với $N=5$, có 7 cách chia nhóm như sau:

- chia thành năm nhóm: mỗi nhóm có 1 người. Mỗi nhóm nhận được 1 chai nước;
- chia thành bốn nhóm: ba nhóm mỗi nhóm có 1 người, một nhóm có 2 người. Mỗi nhóm nhận được 2 chai nước;
- chia thành ba nhóm: hai nhóm mỗi nhóm có 1 người, một nhóm có 3 người. Mỗi nhóm nhận được 3 chai nước;
- chia thành ba nhóm: một nhóm có 1 người, hai nhóm mỗi nhóm có 2 người. Mỗi nhóm nhận được 2 chai nước;
- chia thành hai nhóm: một nhóm có 1 người, một nhóm có 4 người. Mỗi nhóm nhận được 4 chai nước;
- chia thành hai nhóm: một nhóm có 2 người, một nhóm có 3 người. Mỗi nhóm nhận được 6 chai nước;
- chia thành một nhóm có 5 người. Mỗi nhóm nhận được 5 chai nước.

Vậy 6 là số lượng chai nước nhiều nhất mà mỗi nhóm có thể nhận được.

Yêu cầu: Hãy xác định số lượng chai nước nhiều nhất mà mỗi nhóm học sinh có thể nhận được.

Dữ liệu: Cho 10 giá trị của N : 50, 500, 1002, 2003, 3005, 4119, 5555, 9050, 15000, 50000.

Kết quả: Ghi ra file văn bản CHIANHOM.OUT gồm 10 dòng, mỗi dòng chứa số lượng chai nước nhiều nhất tìm được tương ứng với một giá trị của N theo trình tự liệt kê trên.

Lưu ý: Thí sinh không phải nộp file chương trình mà chỉ nộp file **CHIANHOM.OUT**

Năm 2006

Bài 1. Phân cụm

Phân cụm là một bài toán có ý nghĩa ứng dụng quan trọng trong các lĩnh vực như khai phá dữ liệu, thu thập dữ liệu và đòi hỏi phân hoạch tập các điểm dữ liệu ra thành các nhóm sao cho các điểm trong cùng một nhóm là “gần nhau” và “cách xa” các nhóm khác. Trong bài này chúng ta xét một dạng đơn giản của bài toán phân cụm. Cho tập gồm n đối tượng $X = \{x_1, x_2, \dots, x_n\}$, khoảng cách $d(x_i, x_j)$ giữa mọi cặp $x_i \neq x_j$ và một số nguyên dương k ($k \leq n$). Giả thiết rằng: $d(x_i, x_j)$ là các số nguyên dương, $d(x_i, x_j) = d(x_j, x_i)$ và $d(x_i, x_i) = 0$, với mọi $i, j = 1, 2, \dots, n$. Ta gọi một cách phân cụm là một cách phân hoạch tập X ra thành k tập con khác rỗng (mỗi tập con như vậy được gọi là một cụm). Cho $C = \{C_1, C_2, \dots, C_k\}$ là một cách phân cụm, ta gọi **độ phân tách** của cách phân cụm C (ký hiệu là $\rho(C)$) là giá trị nhỏ nhất trong số các khoảng cách giữa hai phần tử bất kỳ thuộc hai cụm khác nhau, nghĩa là

$$\rho(C) = \min \{d(u, v) : u \in C_p, v \in C_q, p \neq q\}.$$

Yêu cầu: Tìm cách phân cụm với độ phân tách là lớn nhất.

Dữ liệu: Vào từ file văn bản CLUSTER.INP:

- Dòng đầu tiên chứa hai số nguyên n và k .
- Dòng thứ i trong số n dòng tiếp theo ghi các số $d(x_i, x_1), d(x_i, x_2), \dots, d(x_i, x_n), i = 1, 2, \dots, n$.

Các số trên cùng một dòng được ghi cách nhau bởi dấu cách.

Kết quả: Ghi ra file văn bản CLUSTER.OUT độ phân tách của cách phân cụm tìm được.

Ví dụ:

CLUSTER . INP	CLUSTER . OUT
4 3	2
0 1 2 3	
1 0 2 3	
2 2 0 3	
3 3 3 0	

Hạn chế:

- Trong tất cả các test: $1 < k \leq n \leq 200; d(x_i, x_j) \leq 32000, i, j = 1, 2, \dots, n$.
- Có 50% số lượng test với $n \leq 100$.

Bài 2. Các cửa hàng

Chủ chung cư Sao Khuê cho thuê dãy N cửa hàng được đánh số từ 1 đến N từ trái qua phải. Năm đầu tiên giá thuê cửa hàng thứ i là a_i đồng. Theo dòng thời gian, có cửa hàng làm ăn phát đạt, có cửa hàng làm ăn thua lỗ. Chủ các cửa hàng làm ăn phát đạt tìm cách thuê thêm diện tích của cửa hàng liền kề để mở rộng công việc kinh doanh. Như vậy có cửa hàng phải đóng cửa, có cửa hàng được mở rộng. Khi một cửa hàng i phát triển và thuê thêm diện tích của cửa hàng j liền kề nó ($j=i-1$ hoặc $j=i+1$), tiền thuê cửa hàng i sẽ được tính theo qui tắc sau:

- Nếu $a_i \bmod 3 = 0$, thì tiền thuê mới là $a_i + a_j$;
- Nếu $a_i \bmod 3 = 1$, thì tiền thuê mới là $2a_i + (a_j \text{ div } 2)$;
- Nếu $a_i \bmod 3 = 2$, thì tiền thuê mới là $4a_i + (a_j \text{ div } 4)$.

Theo luật của chủ chung cư Sao Khuê, mỗi một năm chỉ có không quá một cửa hàng được phép mở rộng và khi có cửa hàng mở rộng dãy các cửa hàng được đánh số lại bắt đầu từ 1 từ trái qua phải. Sau $N-1$ năm, chủ chung cư Sao Khuê giật mình khi phát hiện ra rằng chỉ còn lại duy nhất một cửa hàng ở chung cư. Để tìm hiểu quá trình phát triển của cửa hàng, chủ chung cư đã tìm đến sổ sách. Đáng tiếc là do sự bất cẩn của kế toán, sổ sách ghi nhận chi tiết đã bị thất lạc, chỉ còn lại giá thuê các cửa hàng trong năm đầu tiên và năm thứ $N-1$.

Yêu cầu: Hãy giúp chủ chung cư Sao Khuê khôi phục thông tin về quá trình phát triển của cửa hàng hiện nay.

Dữ liệu: Vào từ file văn bản SHOPS.INP:

- Dòng đầu tiên ghi số N ;
- Dòng thứ hai chứa dãy số nguyên dương a_1, a_2, \dots, a_N ;
- Dòng thứ ba chứa số nguyên là số tiền thuê cửa hàng hiện nay.

Kết quả: Ghi ra file văn bản SHOPS.OUT gồm $N-1$ dòng, dòng thứ i chứa hai số nguyên theo thứ tự là chỉ số của cửa hàng mở rộng và cửa hàng đóng cửa trong năm thứ i , $i=1, 2, \dots, N-1$. (Nếu có nhiều cách khôi phục, chỉ cần đưa ra một cách tùy ý.)

Ví dụ:

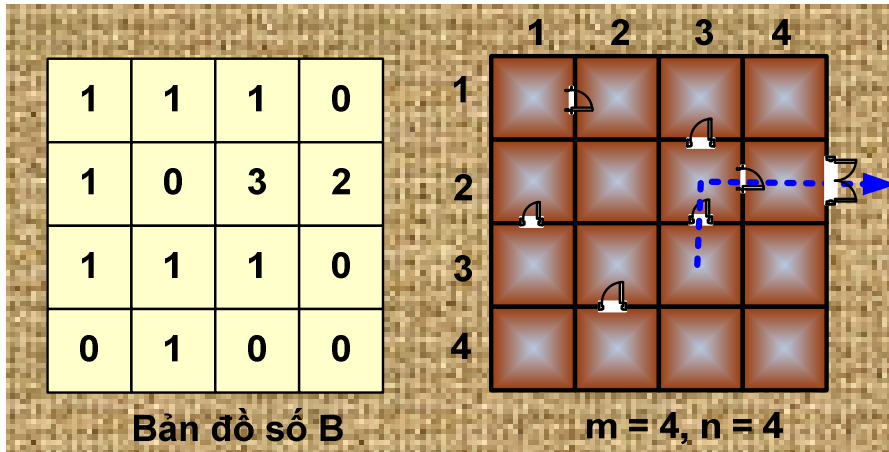
SHOPS . INP	SHOPS . OUT
3	1 2
3 2 4	1 2
21	

Hạn chế:

- Giả thiết là dữ liệu đảm bảo luôn có ít nhất một cách khôi phục.
- Trong tất cả các test: $1 \leq N \leq 30$; $a_i \leq 32000$, $i=1, 2, \dots, N$.
- Có 50% số lượng test với $N \leq 15$.

Bài 3. Thoát mê cung

Mê cung có dạng lưới ô vuông hình chữ nhật kích thước $m \times n$, các dòng của lưới được đánh số từ 1 đến m từ trên xuống dưới, các cột được đánh số từ 1 đến n từ trái sang phải. Ô nằm trên giao của dòng i và cột j được gọi là ô (i, j) . Mỗi ô là một phòng. Vách ngăn giữa hai phòng hoặc giữa phòng với phần bên ngoài có thể có hoặc không có cửa thông nhau. Mê cung được mô tả bởi bản đồ số B là bảng $m \times n$ số nguyên, trong đó thành phần ở vị trí giao của dòng i với cột j là B_{ij} ($0 \leq B_{ij} \leq 4$) cho biết số vách ngăn có cửa của ô (i, j) . Thời gian đi từ một ô sang ô bên cạnh hoặc ra ngoài là 1, nếu vách ngăn tương ứng có cửa.



Yêu cầu: Cho biết m, n , bản đồ số B và u, v – tọa độ một ô nào đó trong mê cung. Hãy xác định một cách đặt cửa cho các phòng đảm bảo thỏa mãn bản đồ B và thời gian T để thoát ra khỏi mê cung từ phòng (u, v) là ít nhất. Biết rằng dữ liệu bản đồ số B đảm bảo có ít nhất một cách đặt cửa để thoát được ra ngoài.

Dữ liệu: Vào từ file văn bản ESCAPE.INP:

- Dòng đầu tiên chứa 4 số nguyên m, n, u và v ;
- Dòng thứ i trong m dòng tiếp theo chứa n số nguyên $B_{i1}, B_{i2}, \dots, B_{in}$.

Các số trên cùng một dòng được phân tách nhau bởi dấu cách.

Kết quả: Đưa ra file văn bản ESCAPE.OUT số nguyên T .

Ví dụ:

ESCAPE . INP			
4	4	3	3
1	1	1	0
1	0	3	2
1	1	1	0
0	1	0	0

ESCAPE . OUT
3

Hạn chế:

- Trong tất cả các test: $2 \leq m, n \leq 50$.
- Có 50% số lượng test với $m, n \leq 25$.

Bài 4. Bumêran

Bumêran là công cụ đi săn bằng đá hoặc gỗ của thổ dân Úc. Nếu ném một cách thích hợp, bumêran sẽ bay tới đích rồi lượn về tay người ném. Để đáp ứng nhu cầu thể thao rèn luyện kỹ năng sử dụng bumêran, một công ty quyết định chế tạo bumêran bằng nhựa. Mỗi khi được ném với một lực nào đó về phía trước, bumêran sẽ bay theo một quỹ đạo nhất định và không nhất thiết quay trở về vị trí ban đầu. Theo thiết kế, mỗi bumêran được gắn với một chuỗi S gồm không quá 250 ký tự lấy từ tập $\{ 'F', 'R' \}$. Quỹ đạo bay của mỗi cách ném bumêran dọc theo đường thẳng đều có thể mô tả được bởi chuỗi ký tự nhận được bằng cách xoá bớt một số ký tự của chuỗi S gắn với nó, trong đó mỗi ký tự cho biết bumêran bay qua một khoảng độ dài 1 về hướng nào:

- F – bumêran bay về phía trước,
- R – bumêran bay ngược trở lại.

Một cách ném bumêran dọc theo đường thẳng được gọi là an toàn nếu thoả mãn 2 điều kiện:

- Điều kiện Bumêran: vị trí bắt đầu và kết thúc của quỹ đạo bay là trùng nhau.
- Điều kiện an toàn: Nếu vị trí ban đầu xuất hiện trên quỹ đạo bay và quá trình bay còn chưa kết thúc thì hướng bay tiếp theo phải là 'F'.

Hai cách ném gọi là khác nhau nếu hai chuỗi mô tả quỹ đạo bay của chúng là khác nhau. Với mỗi bumêran, có thể có nhiều cách ném an toàn khác nhau. Ví dụ, với bumêran có chuỗi $S = 'FFFRRRRRF'$, ta có 3 cách ném an toàn: 'FR', 'FFRR' và 'FFFRRR'.

Lưu ý là trong quỹ đạo bay của bumêran vị trí ban đầu có thể xuất hiện nhiều lần.

Yêu cầu: Cho chuỗi S , hãy xác định số cách ném an toàn khác nhau có thể thực hiện.

Dữ liệu: Vào từ file văn bản BUMERAN.INP chứa chuỗi S .

Kết quả: Đưa ra file văn bản BUMERAN.OUT một số nguyên là số lượng cách ném an toàn có thể thực hiện.

Ví dụ:

BUMERAN . INP	BUMERAN . OUT
FFFRRRRRF	3

Hạn chế: Có 50% số lượng test với chuỗi S có độ dài không quá 100.

Bài 5. Nhóm hình đồng dạng

Trên mặt phẳng cho K hình, đánh số thứ tự từ 1 đến K . Mỗi hình tạo bởi một đường gấp khúc khép kín không tự cắt có các cạnh song song với các trục tọa độ.

Ta nói hai hình là thuộc cùng một nhóm đồng dạng nếu bằng cách thực hiện các phép biến đổi: lấy đối xứng qua trục, xoay, tịnh tiến song song với trục, phóng to, thu nhỏ đối với một hình có thể đặt nó trùng khít lên hình kia.

Yêu cầu: Xác định số lượng nhóm đồng dạng và số lượng hình trong mỗi nhóm tìm được.

Dữ liệu: Vào từ file văn bản GROUP.INP:

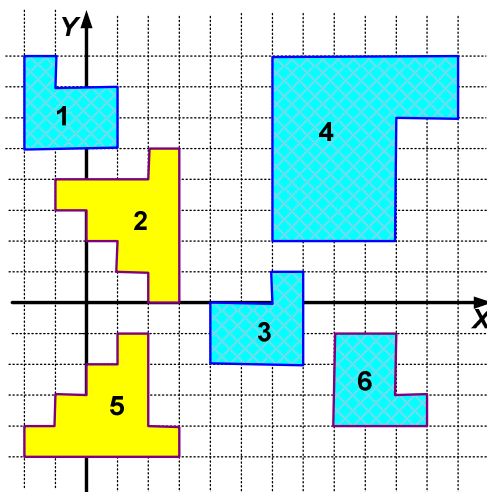
- Dòng đầu tiên ghi số nguyên K là số lượng hình;
 - Dòng thứ i trong K dòng tiếp theo mô tả hình thứ i : đầu tiên là k_i ($4 \leq k_i \leq 200$) là số lượng đỉnh của hình thứ i ($1 \leq i \leq K$), tiếp theo là k_i cặp tọa độ của các đỉnh được liệt kê theo một chiều đi vòng quanh hình (các tọa độ là các số nguyên có trị tuyệt đối không quá 20000).
- Các số trên cùng một dòng được ghi cách nhau một dấu cách.

Kết quả: Ghi ra file văn bản GROUP.OUT:

- Dòng đầu tiên ghi số nguyên M là số lượng nhóm hình đồng dạng tìm được.
- Dòng thứ hai ghi dãy M số nguyên là dãy số lượng các phần tử của các nhóm được liệt kê theo thứ tự không giảm.

Ví dụ:

GROUP . INP	GROUP . OUT
6	2
6 -2 8 -1 8 -1 7 1 7 1 5 -2 5	2 4
12 2 5 3 5 3 0 2 0 2 1 1 1 1 2 0 2 0 3 -1 3 -1 4 2 4	
6 7 -2 4 -2 4 0 6 0 6 1 7 1	
6 10 6 12 6 12 8 6 8 6 2 10 2	
12 1 -1 1 -2 0 -2 0 -3 -1 -3 -1 -4 -2 -4 -2 -5 3 -5 3 -4 2 -4 2 -1	
6 8 -1 8 -4 11 -4 11 -3 10 -3 10 -1	



Hạn chế:

- Trong tất cả các test: $1 \leq K \leq 200$.
- Có 50% số lượng test với $K \leq 100$.

Bài 6. Công viên Disneyland

Bờm và Cuội được mời đến vui chơi miễn phí tại công viên Disneyland nhân dịp công viên này được khai trương tại thành phố Mặt trăng. Trong công viên có tất cả n tụ điểm vui chơi (được đánh số từ 1 đến n). Trước khi đi Bờm và Cuội đã hứa sẽ chụp ảnh tất cả các tụ điểm có trong công viên để giới thiệu cho các bạn cùng lớp. Vì số lượng tụ điểm quá lớn mà thời gian lại hạn hẹp, Bờm và Cuội quyết định thực hiện việc dạo qua tất cả các tụ điểm theo qui tắc sau đây:

- Cùng xuất phát tại tụ điểm 1.
- Mỗi tụ điểm phải được ít nhất một người dạo qua.
- Mỗi người sẽ thăm các tụ điểm theo thứ tự tăng dần của chỉ số.
- Khi kết thúc, mỗi người từ tụ điểm cuối cùng trên đường đi của mình phải quay về nơi xuất phát mà không được đi qua bất cứ tụ điểm nào khác.

Yêu cầu: Cho biết t_{ij} là thời gian đi từ tụ điểm i đến tụ điểm j ($i, j = 1, 2, \dots, n$), hãy xác định giúp Bờm và Cuội cách thực hiện cách đi tuân thủ qui tắc đã nêu sao cho tổng thời gian mà hai người phải đi là nhỏ nhất.

Dữ liệu: Vào từ file văn bản DISNEY.INP:

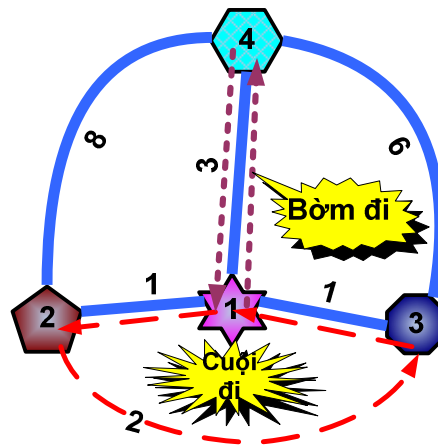
- Dòng đầu tiên chứa số nguyên dương n .
- Dòng thứ i trong số n dòng tiếp theo ghi các số nguyên $t_{i1}, t_{i2}, \dots, t_{in}, i = 1, 2, \dots, n$.

Các số trên cùng một dòng được ghi cách nhau bởi dấu cách.

Kết quả: Ghi ra file văn bản DISNEY.OUT giá trị của tổng thời gian nhỏ nhất tìm được.

Ví dụ:

DISNEY . INP	DISNEY . OUT
4	10
0 1 1 3	
1 0 2 8	
1 2 0 6	
3 8 6 0	



Hạn chế:

- Trong tất cả các test: $1 \leq n \leq 200; 0 \leq t_{ij} \leq 32000, i, j = 1, 2, \dots, n$.
- Có 50% số lượng test với $n \leq 100$.

Năm 2007

Bài 1. Kiểm soát giao thông

Để kiểm soát chặt chẽ việc chấp hành luật giao thông trong đường hầm trên đường cao tốc Bắc – Nam (hình 1), người ta đặt các camera ở trước đường vào và sau đường ra của đường hầm. Các camera này ghi nhận biển số xe đi vào và biển số xe đi ra. Thông tin các xe đi vào hầm theo chiều Bắc – Nam được ghi vào file văn bản NS1.INP, thông tin các xe đi ra khỏi hầm theo chiều Bắc – Nam được ghi vào file văn bản NS2.INP. Các file này định kỳ được truyền tới trạm kiểm soát, nơi mọi ô tô đi ra khỏi đường hầm đều sẽ qua.

Dựa vào thông tin ghi trong hai file này người ta có thể xác định chính xác các xe phạm luật “*tuyệt đối cấm vượt nhau trong đường hầm*”. Các xe đều chạy liên tục, không có xe nào dừng lại trong đường hầm. Độ dài đường hầm cho phép chứa tối đa K xe ($1 \leq K \leq 10000$).

Hãy xác định số lượng các xe vi phạm.

Dữ liệu vào từ 2 file văn bản NS1.INP và NS2.INP mỗi file có nội dung:

- Dòng đầu chứa số K ,
- Mỗi dòng tiếp theo của file chứa một biển số xe là một chuỗi gồm không quá 10 ký tự và không có 2 biển số nào trùng nhau trong file này.

Trong mỗi file có không quá 500000 dòng.

Kết quả: ghi ra file văn bản NS.OUT một số nguyên duy nhất cho biết số lượng xe vi phạm tìm được.

Ví dụ:

NS1 . INP	NS2 . INP	NS . OUT
3	3	2
29A3642	53B1234	
53B1234	29A3642	
NG6677-	15C6688	
58	NG6677-58	
SH09-456	SH09-456	
15C6688		



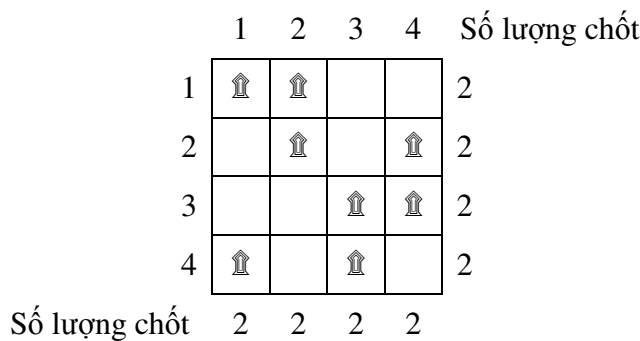
Hình

Bài 2. Phương án phòng thủ

Để phòng vệ cho một trận địa có dạng lưới ô vuông $m \times n$, Bộ chỉ huy quân sự đã tính toán về việc thiết lập các chốt phòng thủ theo khu vực, mỗi chốt sẽ được đặt tại một trong các ô vuông trên trận địa và mỗi ô vuông có không quá một chốt. Giả sử các hàng ngang được đánh số từ 1 đến m theo trình tự từ trên xuống dưới và hàng dọc được đánh số từ 1 đến n theo trình tự từ trái sang phải. Ô nằm trên giao của hàng ngang i và hàng dọc j được gọi là ô (i,j) .

Hệ thống phòng thủ cần phải bảo đảm mỗi hàng ngang, hàng dọc đều có đủ số lượng chốt cần thiết. Bên cạnh đó, dựa vào thông tin tình báo, các chuyên gia cũng tính toán được khả năng đánh phá của địch p_{ij} ($p_{ij} < 1000$) vào từng ô (i,j) trên trận địa (p_{ij} càng lớn nghĩa là khả năng địch đánh phá ô (i,j) càng cao). Căn cứ vào đó, người chỉ huy sẽ bố trí các chốt phòng thủ.

Bạn được mời làm cố vấn quân sự, hãy xác định phương án đặt vị trí các chốt phòng thủ trên trận địa đảm bảo được các yêu cầu về số lượng chốt phòng thủ đồng thời sao cho tổng khả năng đánh phá của địch vào các vị trí được chọn là nhỏ nhất.



Hình 2.

Dữ liệu vào từ file văn bản DEFEND.INP:

- Dòng đầu tiên chứa 2 số nguyên dương m, n ($m, n \leq 100$),
- Dòng thứ 2 chứa m số nguyên không âm lần lượt là số lượng chốt cần có từ hàng ngang thứ 1 đến hàng ngang thứ m ,
- Dòng thứ 3 chứa n số nguyên không âm lần lượt là số lượng chốt cần có từ hàng dọc thứ 1 đến hàng dọc thứ n ,
- m dòng tiếp theo, mỗi dòng chứa n số nguyên không âm mô tả các giá trị p_{ij} .

Các số trên cùng một dòng cách nhau bởi một dấu cách.

Kết quả: ghi ra file văn bản DEFEND.OUT duy nhất số -1 trong trường hợp không có phương án lập hệ thống chốt phòng thủ. Ngược lại, ghi ra số nguyên là tổng khả năng đánh phá của địch vào các vị trí được chọn tương ứng với phương án tìm được.

Ví dụ:

DEFEND . INP	DEFEND . OUT
4 4	22
2 2 2 2	
2 2 2 2	
1 1 8 8	
9 1 10 2	
5 5 2 3	
2 4 10 10	

Hình 2 minh họa cho ví dụ, trong đó chỉ ra cách chọn vị trí đặt chốt phòng thủ đảm bảo các yêu cầu về số lượng chốt phòng thủ với tổng khả năng đánh phá của địch là 22.

Bài 3. Tư vấn

Có N trung tâm tư vấn (đánh số từ 1 đến N), mỗi trung tâm có khả năng tư vấn về một số vấn đề. Trong một ngày làm việc, mỗi trung tâm chỉ tiếp được một số lượng khách nhất định. Một ngày cuối năm, có M khách (đánh số từ 1 đến M), mỗi khách cần tư vấn về một vấn đề. Khách chỉ có thể đến trung tâm có khả năng tư vấn về vấn đề của mình.

Bạn hãy xác định giúp cần ít nhất bao nhiêu trung tâm hoạt động để có thể tư vấn cho tất cả các khách trong ngày cuối năm?

Dữ liệu vào từ file văn bản `ADVICE.INP` gồm:

- Dòng đầu ghi 2 số M ($1 \leq M \leq 500$), N ($1 \leq N \leq 50$),
- Dòng tiếp ghi M chữ cái in hoa liên tiếp, trong đó chữ thứ i mô tả vấn đề cần tư vấn của khách i ,
- Dòng thứ j trong N dòng tiếp theo, ghi thông tin của trung tâm j , gồm một số nguyên là lượng khách cho phép tư vấn trong ngày (không quá 50), sau đó là một dấu cách, tiếp sau là một dãy các chữ cái in hoa liền nhau mô tả dãy vấn đề mà trung tâm có thể tư vấn.

Các số trên một dòng cách nhau bởi một dấu cách. Dữ liệu vào luôn đảm bảo tất cả các khách đều được tư vấn.

Kết quả: ghi ra file `ADVICE.OUT` một số nguyên dương là số trung tâm ít nhất tìm được.

Ví dụ:

<code>ADVICE . INP</code>	<code>ADVICE . OUT</code>
8 4 BNFQISNS 40 QIC 35 UPSF 45 FPHBU 15 WPSCNG	3

Bài 4: Điều khiển HDD

Xét bài toán tìm hành trình di chuyển của đầu đọc đĩa cứng (hình 3). Đầu đọc của đĩa di chuyển dọc theo đường thẳng L . Để thực hiện các yêu cầu đọc đĩa, đầu đọc cần di chuyển đến các vị trí được xác định bởi tập gồm n điểm trên L . Thời gian di chuyển của đầu đọc tỷ lệ với khoảng cách di chuyển. Khi đầu đọc đến đúng điểm đọc, thời gian đọc có thể bỏ qua (bởi vì tốc độ quay của đĩa là nhanh hơn rất nhiều so với tốc độ di chuyển của đầu đọc).

Bài toán có thể phát biểu như sau: Cho n điểm khác nhau trên trục x_1, x_2, \dots, x_n ($x_i \neq 0$). Đầu đọc bắt đầu di chuyển từ điểm gọi **hành trình** là một cách di chuyển đầu đọc đi qua tất cả điểm x_i . **Độ trễ** của điểm x_i trên hành trình được định nghĩa là khoảng cách mà đầu đọc phải di chuyển trên hành trình để lần tiên đạt đến x_i .



Hình 3

trục số
0. Ta
các
tổng
đầu

Hãy xác định hành trình với tổng độ trễ của tất cả các điểm x_1, \dots, x_n **càng nhỏ càng tốt**.

Dữ liệu vào từ file văn bản HDD.INP:

- Dòng đầu tiên ghi số nguyên n ($1 \leq n \leq 100$);
- Dòng thứ hai chứa các số nguyên x_1, x_2, \dots, x_n (từng đôi một khác nhau và khác 0).

Hai số liên tiếp trên cùng một dòng được ghi cách nhau bởi dấu cách. Mỗi số x_i có trị tuyệt đối không vượt quá 32000.

Kết quả: Ghi ra file văn bản HDD.OUT một dòng gồm n số nguyên (cách nhau bởi 1 dấu cách) mô tả lần lượt các vị trí đầu đọc đi qua trong hành trình tìm được.

Ví dụ:

HDD.INP
3
2 1 -1

HDD.OUT
-1 1 2

Gọi độ trễ của điểm x là $dt(x)$.

Hành trình tìm được trong ví dụ có tổng độ trễ là $dt(-1) + dt(1) + dt(2) = 1 + 3 + 4 = 8$

Bài 5: Cụm số không

Số 0 có vai trò rất quan trọng. Tuy vậy, trong các chữ số, số 0 có tuổi đời trẻ nhất. Lần đầu tiên số 0 như một chữ số độc lập được dùng trong lịch của người Maia vào khoảng năm 36 trước công nguyên. Trước đó, khoảng năm 1500 trước công nguyên, người Ba-bi-lon đã sáng tạo ra hệ đếm cơ số 60 nhưng ở đó số 0 chỉ dùng trong khung cảnh nhất định chứ chưa phải là một chữ số độc lập. Lịch sử ra đời đầy huyền bí và hấp dẫn của số 0 đã được chọn làm đề tài trong một cuộc thi hùng biện tiếng Anh.

Trong báo cáo của mình, một thí sinh nói đến vai trò các dãy số 0 trong biểu diễn nhị phân của các số nguyên dương. Dãy bit 0 liên tiếp nằm giữa hai bit 1 hoặc dãy bit 0 ở cuối được gọi là cụm số 0. Ví dụ, số 10100_2 chứa 3 số 0 và chúng được chia thành 2 cụm: một cụm chỉ chứa một số 0, còn cụm kia chứa hai số 0; số 1010010_2 chứa bốn số 0 và chúng được chia thành 3 cụm: hai cụm một số 0 và một cụm hai số 0.

Với hai số nguyên m và n ($0 < m \leq n < 2^{31}$), gọi $B(m,n)$ là tập các biểu diễn nhị phân của tất cả các số nguyên k (biểu diễn của mỗi số chỉ gồm các chữ số có nghĩa của nó) thoả mãn điều kiện: $m \leq k \leq n$. Ví dụ, với $m = 5$ và $n = 14$ ta có tập $B(5,14)$ gồm các phân tử sau:

- | | |
|-------------|-------------|
| 101 | 1010 |
| 110 | 1011 |
| 111 | 1100 |
| 1000 | 1101 |
| 1001 | 1110 |

Gọi S_i là số lượng cụm có i số 0. Với tập $B(5,14)$ ta có thể thống kê số lượng các cụm 0 của các phân tử trong nó và được kết quả $S_1 = 7, S_2 = 2, S_3 = 1$ và $S_i = 0$ với mọi $i > 3$.

Kết thúc báo cáo của mình, thí sinh đề xuất với các bạn một bài toán nhỏ:

Cho biết m và p giá trị $S_{i_1}, S_{i_2}, \dots, S_{i_p}$ ($1 \leq i_j \leq 30, j = 1, 2, \dots, p; 1 \leq p \leq 30$). Hãy xác định giá trị n nhỏ nhất, lớn hơn hoặc bằng m sao cho tập $B(m,n)$ cho kết quả thống kê số lượng cụm có i_j số 0 bằng S_{i_j} ($j = 1, 2, \dots, p$) đã cho.

Dữ liệu: Vào từ file văn bản ZGROUP.INP gồm nhiều test, mỗi test cho trên một nhóm dòng:

- Dòng thứ nhất trong nhóm chứa hai số nguyên m và p ,
- Dòng thứ j trong p dòng tiếp theo chứa 2 số nguyên i_j và S_{i_j} .

Các số trên một dòng cách nhau bởi một dấu cách.

Kết quả: Ghi ra file văn bản ZGROUP.OUT, kết quả mỗi test đưa ra trên một dòng chứa số nguyên n tìm được tương ứng với test.

Ví dụ:

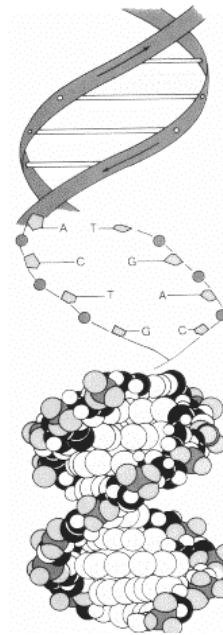
ZGROUP . INP	
5	1
2	2
5	2
1	7
2	2

ZGROUP . OUT
12
14

Bài 6. α -DNA

Khi khám phá ra sự sống trên hành tinh Alpha, các nhà khoa học bắt đầu tiến hành nghiên cứu cấu trúc của các cơ thể sống. Họ ngạc nhiên khi thấy chúng có nhiều điểm khác với sự sống trên Trái đất.

Các cơ thể sống có cấu thành từ các phần tử cơ bản được các nhà đặt tên là α -DNA (hình 4). Có rất nhiều loại α -DNA, mỗi α -DNA là các thành tố cơ bản gọi là mã β . Chiều dài của một α -DNA là một số dương tương ứng với số lượng mã β trong chuỗi. Trong quá trình của một loài sinh vật, chiều dài α -DNA của loài đó có thể sẽ lớn lên số của chiều dài ban đầu. Như vậy, hai α -DNA được gọi là có quan tiếp nếu chiều dài của α -DNA này là bội số của chiều dài α -DNA



khoa học một chuỗi nguyên phát triển bằng bội hệ trực kia. cho việc

nhiều nhất hệ trực

Hình 4.

Hãy giúp các nhà khoa học chọn ra từ tập mẫu đã cho một số lượng các α -DNA sao cho không có hai α -DNA nào được chọn ra có quan tiếp với nhau.

Dữ liệu vào từ file văn bản ADNA.INP:

- Dòng đầu tiên chứa số nguyên dương n , là số lượng các α -DNA trên hành tinh Alpha mà các nhà khoa học đã thu thập được ($1 \leq n \leq 200$).
- Dòng 2: chứa n số nguyên dương $a_1, a_2, a_3, \dots, a_n$ cho biết chiều dài của các α -DNA tương ứng ($1 \leq a_i \leq 10^9$).

Các số trên cùng một dòng cách nhau bởi một dấu cách.

Kết quả: ghi ra file văn bản ADNA.OUT một số nguyên duy nhất là số lượng α -DNA chọn được.

Ví dụ:

ADNA . INP	ADNA . OUT
8 1 2 3 5 6 8 7 9	5

Năm 2008

Bài 1. Mật khẩu

Để bảo vệ máy tính của mình, việc hạn chế người khác thâm nhập là một vấn đề đặt ra cho nhiều người sử dụng máy tính. Để thuận tiện trong lưu trữ và sử dụng, một người đã qui định mật khẩu mà mình chọn là một xâu T gồm các chữ cái la tinh thường thỏa mãn tất cả các tính chất sau:

- Xâu T có độ dài m là một số chẵn,
- Các ký tự trong xâu T được đánh số từ trái sang phải bắt đầu từ 1 và thoả điều kiện:
 - $T_i = T_{m-i+1}$ với mọi i là số lẻ ($1 \leq i \leq m \text{ div } 2$),
 - $T_i \neq T_{m-i+1}$ với mọi i là số chẵn ($1 \leq i \leq m \text{ div } 2$),

Ví dụ:

- “test” là một mật khẩu có độ dài 4.
- “window” không phải là mật khẩu vì ký tự thứ 3 là ‘n’ khác ký tự thứ 4 (=6-3+1) là ‘d’.

Nhằm không để lộ mật khẩu, anh ta đã dấu mật khẩu T trong một xâu S chỉ chứa các chữ cái la tinh thường. T là xâu có thứ tự từ điển nhỏ nhất trong số các xâu con dài nhất của S thoả mãn điều kiện đã nêu trên của mật khẩu. Xâu con của một xâu là xâu thu được từ nó bằng cách xóa đi một số ký tự nào đó.

Yêu cầu: Cho xâu S . Tìm mật khẩu T đã dấu trong S .

Dữ liệu: Vào từ file văn bản PASSWORD.INP:

- Dòng đầu tiên chứa số nguyên n là độ dài của xâu S ($1 \leq n \leq 2000$),
- Dòng thứ hai chứa xâu S có độ dài n .

Kết quả: Ghi ra file văn bản PASSWORD.OUT chứa mật khẩu tìm được

Ví dụ:

PASSWORD.INP	PASSWORD.OUT
6 window	wdow

Bài 2. Tảo biển

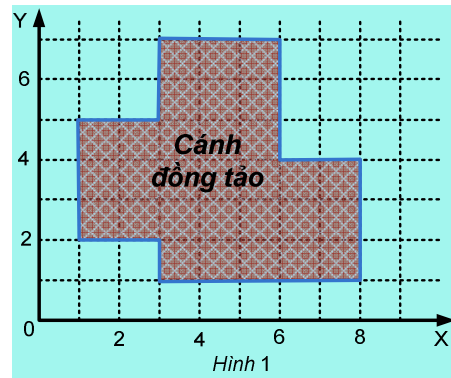
Tảo biển đóng một vai trò hết sức quan trọng trong quá trình đảm bảo cân bằng sinh thái trên trái đất. Khả năng hấp thụ CO₂ của nó vượt xa khả năng tương ứng ở tất cả các cánh rừng nhiệt đới trên thế giới. Tảo là thức ăn chủ yếu của nhiều loại sinh vật biển và một số tảo cũng có mặt trên bàn ăn của con người. Vì vậy, sự phân bố và phát triển của tảo trên bề mặt đại dương được theo dõi chặt chẽ từ ngoài không gian. Các vệ tinh sinh thái thường xuyên chụp và gửi về trái đất bản đồ vùng tảo phát triển. Mỗi vùng có mật độ tảo cao được gọi là cánh đồng tảo. Một số thông tin về các cánh đồng tảo được Liên hợp quốc bán cho các quốc gia quan tâm ở ba mức:

- Mức 1 (Đầy đủ): bao gồm dữ liệu và hệ thống xử lý,
- Mức 2 (Kết quả): thông tin được xử lý và kết quả được chuyển giao cho người mua,
- Mức 3 (Thông tin sơ cấp): người mua nhận được thông tin ở dạng sơ cấp, chưa qua xử lý.

Dĩ nhiên giá bán ở mức 3 là thấp nhất. Công ty SEAFOOD có đội tàu viễn dương khá lớn nhưng cũng chỉ đủ kinh phí mua thông tin mức 3. Thông tin mà Công ty mua là bản đồ cánh đồng tảo ở khu vực nhiệt đới Thái Bình Dương có dạng là một miền được bao bởi đường gấp khúc kín không tự cắt, có các cạnh song song với trục tọa độ và được xác định bởi tọa độ các đỉnh của nó. Các đỉnh được liệt kê theo một chiều nào đó đi vòng quanh đường gấp khúc và không có 3 đỉnh liên tiếp nào thẳng hàng. Tọa độ các đỉnh đều là những số nguyên không âm, nhưng cho ở dạng nguyên thủy: giá trị các tọa độ của hai điểm liên tiếp được viết liên tục trên một dòng không có ký tự phân cách tạo thành một xâu ký tự số.

Ví dụ, bản đồ tảo ở hình 1 có tọa độ các đỉnh là (1,2), (1,5), (3,5), (3,7), (6,7), (6,4), (8,4), (8,1), (3,1) và (3,2). Nếu các đỉnh được liệt kê theo chiều kim đồng hồ thì thông tin Công ty nhận được là:

1215
 3537
 6764
 8481
 3132



Mỗi phòng của Công ty cần một số loại thông tin. Bạn là nhân viên của Phòng Kế hoạch và ở đó người ta yêu cầu bạn cho biết chu vi d và diện tích S của cánh đồng. Nhà cung cấp thông tin cho biết số đỉnh của đường gấp khúc không quá 200000, giá trị tọa độ mỗi đỉnh không vượt quá 10^9 . Dữ liệu đảm bảo chính xác.

Dữ liệu: Vào từ file văn bản SEAWEED.INP gồm:

- Dòng đầu tiên chứa số nguyên dương n cho biết số đỉnh của đường gấp khúc,
- Mỗi dòng trong các dòng tiếp theo chứa một xâu chỉ chứa các ký tự số thập phân ứng với một dòng thông tin về các đỉnh của đường gấp khúc mà công ty đã nhận được. Thông tin về các đỉnh được liệt kê theo một chiều nào đó (theo chiều kim đồng hồ hoặc ngược lại).

Dữ liệu cho đảm bảo tồn tại duy nhất một đường gấp khúc thỏa mãn điều kiện bài toán.

Kết quả: Ghi ra file văn bản SEAWEED.OUT, gồm một dòng chứa 2 số nguyên d và S , ghi cách nhau một dấu cách.

Ví dụ:

SEAWEED.INP	SEAWEED.OUT
10	26 30
1215	
3537	
6764	
8481	
3132	

Bài 3. Nhân của biểu thức

Biểu thức ngoặc là xâu chỉ gồm các ký tự '(' và ')'. Biểu thức ngoặc đúng được định nghĩa một cách đệ qui như sau:

- Biểu thức rỗng là biểu thức ngoặc đúng,
- Nếu A là biểu thức ngoặc đúng thì (A) cũng là một biểu thức ngoặc đúng,
- Nếu A và B là hai biểu thức ngoặc đúng thì AB cũng là một biểu thức ngoặc đúng.

Số lượng ký tự trong biểu thức ngoặc đúng gọi là độ dài của biểu thức.

Cho S là một biểu thức ngoặc đúng độ dài 2n (n > 0). Nhân của S ký hiệu là Ker(S) và được xác định như sau. Gọi U là xâu thu được từ S bằng cách bỏ đi m ký tự cuối và m ký tự đầu của S, trong đó m = n div 2. Khi đó:

- Ker(S) = U, nếu U là biểu thức ngoặc đúng và U ≠ S,
- Ker(S) = ∅ (biểu thức rỗng), trong trường hợp ngược lại,

Qui ước Ker(∅) = ∅.

Ví dụ, S = '()()()', ta có Ker(S) = '()'. Nếu S = '()()', có Ker(S) = ∅, vì U = ')(()' – không phải là biểu thức ngoặc đúng. Dễ dàng nhận thấy, với n = 1, Ker(S) = ∅.

Với biểu thức ngoặc đúng S cho trước ta có thể thực hiện đệ qui nhiều lần phép xác định nhân Ker(Ker(...(Ker(S))...)) cho đến khi nhận được biểu thức rỗng. Ta gọi bậc của biểu thức S là số lần thực hiện đệ qui việc xác định nhân của S cho đến khi nhận được biểu thức rỗng.

Ví dụ, biểu thức S = '()()()' có bậc 3 vì:

$$()()() \rightarrow () \rightarrow () \rightarrow \emptyset.$$

Biểu thức S = '()()' có bậc 1 vì nhân của nó Ker(S) = ∅.

Yêu cầu: Xét các biểu thức ngoặc đúng độ dài 2n và có bậc p. Các biểu thức này được sắp xếp theo thứ tự từ điển ('<') và đánh số bắt đầu từ 1 trở đi. Hãy tìm biểu thức thứ k.

Dữ liệu: Vào từ file văn bản EKERNEL.INP gồm một dòng chứa 3 số nguyên n, p và k, các số cách nhau một dấu cách (1 ≤ n ≤ 40, 1 ≤ p ≤ 6, 1 ≤ k ≤ 3 × 10¹⁸). Dữ liệu đảm bảo tồn tại biểu thức cần tìm.

Kết quả: Ghi ra file văn bản EKERNEL.OUT xâu ký tự xác định biểu thức tìm được.

Ví dụ:

EKERNEL . INP	EKERNEL . OUT
3 1 2	() (())

Bài 4. LED

Để tạo không khí sôi nổi và nhắc nhở học sinh cố gắng học tập, nhà trường lắp một đồng hồ đếm ngược, cho biết còn lại bao nhiêu ngày, bao nhiêu giờ, phút, giây sẽ tới một thời điểm quan trọng chẳng hạn như kiểm tra cuối kỳ của môn văn, thi học sinh giỏi Quốc gia, thi Tốt nghiệp v. v. . . Trên mặt đồng hồ có 4 cửa sổ: cửa sổ D chỉ ngày hiển thị 3 chữ số cho biết số ngày còn lại, cửa sổ H chỉ giờ, cửa sổ M chỉ phút và cửa sổ S chỉ giây. Mỗi cửa sổ H, M và S hiển thị 2 chữ số chỉ giờ, phút, giây còn lại. Mỗi chữ số được hiển thị bằng các Điốt phát sáng (LED) đặt dọc và ngang. Hình 2a cho biết cách hiển thị mỗi chữ số từ 0 đến 9. Ví dụ, để hiển thị số 0 cần 4 LED dọc và 2 LED ngang. Nếu thời gian còn lại là 18 ngày, 10 giờ, 20 phút và 10 giây thì bảng hiển thị của đồng hồ đếm ngược sẽ có dạng như ở hình 2b.

Chữ số	0	1	2	3	4	5	6	7	8	9
Hiển thị										
Số LED dọc	4	2	2	2	3	2	3	2	4	3

Hình 2a



Hình 2b

Như vậy, sau mỗi giây trạng thái của đồng hồ đếm ngược sẽ bị thay đổi và để hiển thị trạng thái này ta phải bật sáng một số lượng xác định các LED dọc và ngang. Để thuận tiện trong việc xử lý, người ta lưu lại tổng số lượng các LED dọc đã được bật sáng qua các lần hiển thị cho đến trạng thái hiện hành. Giả sử lần đầu tiên được bật, bảng đồng hồ đếm ngược chỉ 18 ngày, 10 giờ, 20 phút, 15 giây thì tổng số lượng LED dọc được bật sáng sẽ là 159.

Yêu cầu: Biết trạng thái hiện tại của đồng hồ đếm ngược (các giá trị D, H, M, S) và tổng số lượng các LED dọc đã được bật sáng, hãy xác định trạng thái của đồng hồ đếm ngược khi lần đầu tiên được bật.

Dữ liệu: Vào từ file văn bản LEADS.INP gồm một dòng chứa 5 số nguyên D, H, M, S, T ($0 < T < 2^{31}$) ghi cách nhau ít nhất một dấu cách cho biết thông tin về ngày, giờ, phút, giây của trạng thái hiện hành và tổng số lượng các LED dọc đã được bật sáng.

Kết quả: Ghi ra file văn bản LEADS.OUT trên một dòng 4 số nguyên cho biết trạng thái của đồng hồ khi lần đầu tiên được bật theo thứ tự ngày, giờ, phút và giây.

Ví dụ:

LEADS.INP	LEADS.OUT
18 10 20 10 159	18 10 20 15

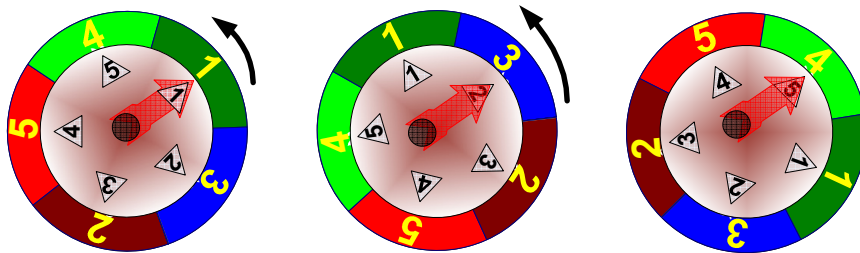
Bài 5. Quay số trúng thưởng

Trong trò chơi quay số trúng thưởng, có một bánh xe chỉ quay quanh trục ngược chiều kim đồng hồ, trên đó có n ô. Trên mỗi ô có ghi một số nguyên dương không vượt quá n và không có hai số nào giống nhau. Số ghi trên ô được gọi là giá trị của ô. Có một kim định vị gắn cố định bên ngoài bánh xe để khi quay, các ô của bánh xe sẽ lần lượt lướt qua kim định vị. Khi bánh xe dừng quay, kim định vị trở vào ô nào ta nói bánh xe dừng ở ô đó.

Lượt chơi của một người được thực hiện như sau. Đầu tiên, người chơi được quyền xoay để bánh xe dừng lại ở một ô xuất phát tùy chọn. Hệ thống sẽ tự động đánh số các ô trên bánh xe từ 1 đến n theo chiều kim đồng hồ bắt đầu từ ô xuất phát này. Các số thứ tự này sẽ được giữ nguyên cho đến khi kết thúc lượt chơi. Sau khi chọn ô xuất phát, trả lời câu hỏi tương ứng với ô này và nhận phần thưởng nếu trả lời đúng, người chơi bấm nút cho bánh xe tự động quay. Hệ thống điều khiển được lập trình để bánh xe sẽ dừng lại ở ô có số thứ tự lớn hơn số thứ tự của ô ở lần dừng trước. Đồng thời, giá trị ở ô mới cũng phải lớn hơn giá trị ở ô cũ. Mỗi lần bánh xe dừng, người chơi sẽ có cơ hội trả lời câu hỏi để nhận phần thưởng. Sau khi trả lời đúng câu hỏi, người chơi lại bấm nút cho bánh xe tiếp tục quay. Lượt chơi kết thúc khi người chơi trả lời sai hoặc hệ thống điều khiển không tìm được ô dừng tiếp theo.

Nếu chọn vị trí xuất phát hợp lý người chơi sẽ có nhiều cơ hội trả lời các câu hỏi để lĩnh thưởng.

Ví dụ, giả sử bánh xe có 5 ô lần lượt mang giá trị 1, 3, 2, 5, 4. Nếu chọn ô xuất phát là ô có giá trị 2, bánh xe có thể dừng ở ô có giá trị 5, 4 hay 3. Dù dừng lại ở ô nào thì lượt chơi cũng kết thúc. Nếu chọn ô xuất phát là ô có giá trị 1, người chơi trong trường hợp may mắn nhất, có thể có tới ba cơ hội trả lời câu hỏi của trò chơi (xem ví dụ trong hình 3).



Hình 3- Nếu bắt đầu từ 1 ta có thể có tối đa 3 cơ hội trả lời

Yêu cầu: Cho n và giá trị của các ô trong vòng tròn. Hãy cho biết số cơ hội nhiều nhất được trả lời câu hỏi của trò chơi mà người chơi có thể chờ đợi nếu chọn ô xuất phát hợp lý.

Dữ liệu: Vào từ file văn bản LWEEL.INP:

- Dòng đầu tiên chứa số nguyên n ($1 < n \leq 1000$) là số ô trên bánh xe.
- Dòng thứ hai chứa n số nguyên dương cho biết lần lượt giá trị của các ô trên bánh xe được liệt kê theo chiều kim đồng hồ bắt đầu từ một ô nào đó. Các số trên cùng một dòng cách nhau bởi ít nhất 1 dấu cách.

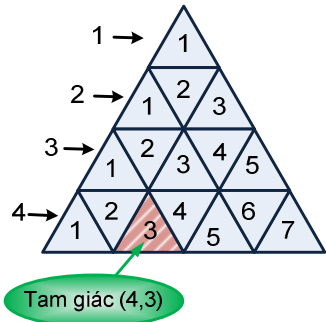
Kết quả: Ghi ra file văn bản LWEEL.OUT một số nguyên dương duy nhất là số cơ hội nhiều nhất được trả lời câu hỏi.

Ví dụ:

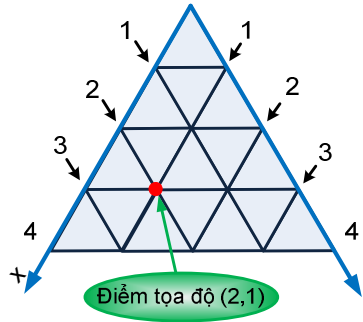
LWEEL . INP	LWEEL . OUT
5	3
1 3 2 5 4	

Bài 6. Công nghệ Nano

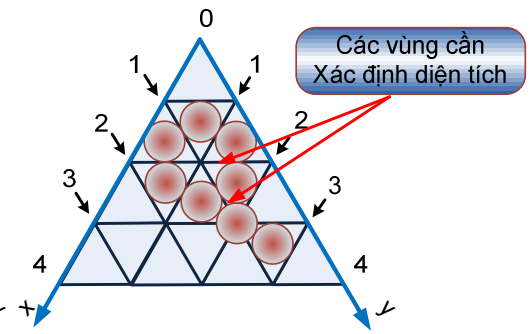
Công nghệ Nano mang lại nhiều thay đổi trong việc chế tạo các mạch điện tử. Ví dụ, các bảng mạch không có dạng hình chữ nhật mà là hình tam giác đều, từ đó tạo ra các con chip hình kim tự tháp. Xét việc chế tạo một bảng mạch Nano. Bảng mạch có hình tam giác đều và được chia thành lưới các tam giác đều con bằng các đường song song với cạnh tam giác. Các tam giác con có độ dài cạnh bằng 1 và tạo thành các hàng đánh số từ 1 trở đi, từ trên xuống dưới, ở mỗi hàng các tam giác được đánh số từ 1 trở đi từ trái sang phải. Một tam giác được xác định bởi hai tọa độ: hàng và vị trí trong hàng (Hình 4a). Các đường song song với hai cạnh bên bảng mạch tạo thành lưới tọa độ, xác định vị trí đỉnh các tam giác con (Hình 4b).



Hình 4a



Hình 4b



Hình 4c

Các linh kiện Nano có dạng hình tròn với kích thước đúng bằng hình tròn nội tiếp tam giác đều với độ dài cạnh là 1. Có 2 loại linh kiện *T* và *P*. Linh kiện loại *T* được cấy gọn vào một tam giác con trong lưới, còn linh kiện loại *P* được cấy sao cho tâm của linh kiện trùng với đỉnh của tam giác con. Mỗi linh kiện đều nằm gọn trong bảng. Sau khi đã cấy các linh kiện, trên bảng mạch xuất hiện những vùng được bao bọc bởi các linh kiện. Những vùng này là những vùng nhạy cảm, cần được làm sạch và phủ một lớp bảo vệ đặc biệt. Để tính chi phí làm sạch và phủ các vùng này, người ta cần biết tổng diện tích của chúng trên mạch đã thiết kế.

Ví dụ, trên bảng mạch ở hình 4c có 7 linh kiện loại *T* được cấy vào các tam giác con (2,1), (2,2), (2,3), (3,2), (3,3), (3,4) và (4,6), có một linh kiện loại *P* cấy vào vị trí đỉnh (1,2) và xuất hiện 2 vùng nhạy cảm.

Yêu cầu: Cho biết *m* – số linh kiện loại *T*, *n* – số linh kiện loại *P*, tọa độ của các linh kiện (tọa độ linh kiện loại *T* là tọa độ tam giác chứa linh kiện, tọa độ linh kiện loại *P* là tọa độ tâm của nó). Hãy xác định diện tích vùng khép kín với độ chính xác 5 chữ số sau dấu chấm thập phân.

Dữ liệu: Vào từ file văn bản NANO.INP:

- Dòng đầu tiên chứa 2 số nguyên không âm *m* và *n* ($0 \leq m+n \leq 20000$),
 - Mỗi dòng trong *m* dòng tiếp theo chứa 2 số nguyên dương xác định tọa độ một linh kiện loại *T*,
 - Mỗi dòng trong *n* dòng tiếp theo chứa 2 số nguyên dương xác định tọa độ một linh kiện loại *P*.
- Các số trên một dòng cách nhau một dấu cách. Giá trị mỗi tọa độ không vượt quá 10^5 .

Kết quả: Đưa ra file văn bản NANO.OUT dưới dạng số thực với 5 chữ số sau dấu chấm thập phân.

Ví dụ:

NANO . INP	NANO . OUT
7 1	0.35586
2 1	
2 2	
2 3	
3 2	
3 3	
3 4	
4 6	
1 2	

Ghi chú: $\pi = 3.14159265358979323846264338327950288419716939937510$