Bai1.cpp; bai2.cpp; bai3.cpp

Bài 1.

Kết quả thi đấu quốc gia của N vận động viên ( đánh số từ 1 đến N ) trên M môn ( đánh số từ 1 đến M ) được đánh giá bằng điểm ( giá trị nguyên không âm ). Với vận động viên, ta biết điểm đánh giá trên từng môn của vận động viên ấy. Các điểm này được ghi trong File văn bản có cấu trúc:

+ Dòng đầu ghi số vận động viên, số môn và số k.

+ Các dòng tiếp theo, mỗi dòng ghi các điểm đánh giá trên tất cả m môn của một vận động viên theo thứ tự môn thi 1, 2, .., m. Các dòng này được ghi theo thứ tự vận động viên 1, 2, .., N

+ Các số ghi trên một dòng cách nhau một dấu cách.

Cần chọn ra k vận động viên và k môn để lập một đội tuyển thi đấu Olympic quốc tế, trong đó mỗi vận động viên chỉ được thi đấu 1 môn ( 1<=k<=M, N), sao cho tổng số điểm của các vận động viên trên các môn đã chọn là lớn nhất.

Yêu cầu:

Đọc bảng điểm từ 1 file văn bản (bai1.inp), sau đó cứ mỗi lần nhận một giá trị k nguyên dương từ bàn phím, chương trình đưa vào file bai1.out kết quả tuyển chọn dưới dạng k cặp (i, j) với nghĩa vận động viên i được chọn thi đấu môn j và tổng số điểm tương ứng với cách đã chọn. Các giá trị giới hạn 1<=M, N<= 20. Điểm đánh giá từ 0 đến 100.

Thí dụ:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| bai1.inp  3 3 1  1 5 0  5 7 4  3 6 3 | bai1.out  7  2 2 | bai1.inp  3 3 2  1 5 0  5 7 4  3 6 3 | bai1.out  11  2 1  3 2 | bai1.inp  3 3 0  1 5 0  5 7 4  3 6 3 | bai1.out  No |

**Bài 2**

Địa bàn hoạt động của một điệp viên là một khu phố mà ở đó chỉ có các đường phố ngang, dọc tạo thành một lưới ô vuông. Với mục đích bảo mật, thay vì tên đường phố, điệp viên đánh số các phố ngang từ 0 đến m và các phố dọc từ 0 đến n. Ở một số ngã ba hoặc ngã tư có các trạm kiểm soát. Anh ta đang đứng ở nút giao của hai đường (i1, j1) (j1 - đường ngang; i1 - đường dọc) và cần tới điểm hẹn ở giao của hai đường (i2, j2). Để tránh bị theo dõi, đường đi phải không qua các trạm kiểm soát và cứ tới chỗ rẽ thì nhất thiết phải đổi hướng đi, thậm chí có thể sang đường và đi ngược trở lại. Việc đổi hướng chỉ được thực hiện ở ngã ba hoặc ngã tư. Hãy xác định đường đi ngắn nhất tới điểm hẹn hoặc cho biết không có đường đi đáp ứng được yêu cầu đã nêu.

***Dữ liệu: vào từ file bai2.INP***

Dòng đầu: m n i1 j1 i2 j2 ( 0 ≤ m, n ≤ 100)

Các dòng sau: mỗi dòng 2 số i, j (toạ độ trạm kiểm soát).

***Kết quả: đưa ra file bai2.OUT***

Dòng đầu: độ dài đường đi ngắn nhất hoặc thông báo NO nếu không có đường đi.

Các dòng sau: mỗi dòng 2 số i, j chỉ nút tiếp theo cần tới theo đường đi tìm được, bắt đầu là i1 j1 và kết thúc là i2 j2.

***Ví dụ:***

|  |  |  |
| --- | --- | --- |
| bai2.INP |  | bai2.OUT |
| 4 5 0 0 5 4  0 1  0 4  2 2  2 3  4 0  5 2  5 3  -1 |  | 13  0 0  1 0  1 1  1 0  2 0  2 1  3 1  3 2  4 2  4 3  3 3  4 3  4 4  5 4 |

**Bài 3**

Có 3N chiếc bánh trên bàn xếp thành một vòng tròn. Tom và Jerry sẽ ăn hết bánh theo qui luật sau: Tom có thể chọn bất cứ chiếc bánh nào để ăn nhưng sau đó Jerry sẽ ăn hai chiếc bánh kề ngay bên cạnh chiếc bánh Tom đã ăn. Sau N lần số bánh sẽ được ăn hết. Kích thước của các chiếc bánh là khác nhau. Hãy viết chương trình giúp Tom tính xem có thể ăn được tối đa số bánh (tính theo tổng kích thước của các bánh đã ăn) là bao nhiêu.

### Input

Dữ liệu của chương trình được cho trong một file text như sau: kích thước của các chiếc bánh là các số nguyên từ 1 tới 100 được ghi trên 1 dòng ngăn cách với nhau bằng một dấu cách. Số bánh luôn là một số nguyên chia hết cho 3 và không quá 48.

### Output

Kết quả xử lý của chương trình ghi vào một file text.

### Ví dụ

|  |  |
| --- | --- |
| **bai3.inp** | **bai3.out** |
| **33 33 34** | **34** |
| **5 17 22 34 18 4** | **51** |
| **11 1 1 85 1 1** | **96** |
| **6 13 14 4 14 10 1 20 18** | **48** |