

MẠCH ĐIỆN

Nhà trường được tài trợ một phòng Tin học. Có tất cả n thiết bị: máy tính, máy in, máy chiếu, . . . Các thiết bị được đánh số từ 1 đến n . Người ta phải dọn gấp một phòng học để lắp máy. Bắt tay vào lắp đặt mọi người mới phát hiện ra là trong phòng chỉ có một ổ điện trên tường. Tuy vậy, đó không phải là vấn đề lớn: k dây cắm kéo dài được mua về và rắc rối bắt đầu xuất hiện: mỗi dây cắm kéo dài có một khả năng chịu tải riêng. Dây cắm thứ i có a_i ổ cắm và chịu được công suất b_i , còn thiết bị thứ j có công suất c_j . Theo quy định phòng cháy – chữa cháy, mỗi dây kéo dài chỉ được cắm tối đa một dây cắm kéo dài khác, tổng công suất tiêu thụ gắn với mỗi dây cắm kéo dài cũng không được vượt quá khả năng chịu tải của dây cắm. Ổ cắm trong phòng có khả năng chịu tải vô hạn. Thiết bị có thể được cắm vào ổ kéo dài hay cắm ngay vào ổ cắm trên tường. Các dây cắm được đánh số từ 1 đến k .

Yêu cầu: Hãy xác định có tồn tại một cách cắm các thiết bị vào các ổ điện hay không và đưa ra thông báo **Yes** hoặc **No**. Trong trường hợp **Yes** – chỉ ra cách cắm: với mỗi dây kéo dài i đưa ra số u_i – nơi dây này được cắm, $u_i=0$ – dây cắm vào ổ trên tường, bằng -1 – không sử dụng. Với thiết bị j đưa ra số v_j – dây cắm nơi cắm thiết bị này. $v_j = 0$ nếu thiết bị cắm vào ổ điện trên tường.

Dữ liệu: Vào từ file văn bản **ELECTRO.INP**:

- Dòng đầu tiên chứa số nguyên k ($1 \leq k \leq 10^5$),
- Dòng thứ i trong k dòng sau chứa 2 số nguyên a_i và b_i ($1 \leq a_i \leq 10^5$, $1 \leq b_i \leq 10^9$),
- Dòng thứ $k+2$ chứa số nguyên n ($1 \leq n \leq 10^5$),
- Dòng cuối cùng chứa n số nguyên c_1, c_2, \dots, c_n ($1 \leq c_j \leq 10^9$, $j = 1 \div n$).

Kết quả: Đưa ra file văn bản **ELECTRO.OUT**:

- Dòng đầu tiên đưa ra thông báo **Yes** hoặc **No**,
- Nếu kết quả là **Yes** thì:
 - Dòng thứ 2 đưa ra **k** số nguyên **u_1, u_2, \dots, u_k** ,
 - Dòng thứ 3 đưa ra **n** số nguyên **v_1, v_2, \dots, v_n** .

Ví dụ:

ELECTRO.INP
2
2 20
2 10
3
10 5 5

ELECTRO.OUT
Yes
0 1
1 2 2

THI LẬP TRÌNH

Steve là cán sự Tin học của lớp. Để giúp các bạn trong lớp chuẩn bị thi Olympic Tin học Steve tải từ internet về một loạt các files, mở nén và lưu vào đĩa cứng trên máy của mình. Số lượng files nhận được là khá nhiều và Steve không tính được, thông tin về bao nhiêu kỳ thi đã được tải về.

Cặp files được gọi là một test nếu chúng cùng nằm trong một thư mục, có tên là “***XY***” và “***XY.a***”, trong đó “***XY***” là một số nguyên – số hiệu test (chữ số đầu tiên là 0 nếu số hiệu bé hơn 10). File thứ nhất – dữ liệu vào, file thứ 2 – kết quả.

Một thư mục được gọi là thư mục ***test*** nếu nó chứa tất cả các ***tests*** từ 1 đến ***n*** ($1 \leq n \leq 99$), ngoài ra không chứa file nào khác (nhưng có thể có các thư mục con nào đó).

Thư mục gọi là *Bài toán* nếu nó chứa file có tên là “***check***” với phần mở rộng bất kỳ (có thể là rỗng) và chứa thư mục con “***tests***” – thư mục ***test***. Trong thư mục đầu bài có thể còn có các file và thư mục con khác.

Thư mục được gọi là *mô tả kỳ thi* nếu nó có ít nhất một thư mục con và các thư mục con của nó đều là bài toán.

Yêu cầu: Cho biết tất cả tên đầy đủ tất cả các files mà Steve đã tải về. Hãy xác định có bao nhiêu kỳ thi được tải về.

Dữ liệu: Vào từ file văn bản **CONTESTS.INP**:

- Dòng đầu tiên chứa số nguyên ***n*** – số lượng files ($1 \leq n \leq 1\,000$),
- Mỗi dòng trong ***n*** dòng tiếp theo chứa một tên file đầy đủ có không quá 200 ký tự, tên của ổ đĩa là một ký tự trong phạm vi từ “***A***” đến “***Z***”, các ký tự trong

tên file hoặc thư mục có mã trong phạm vi [33, 126], trong tên đầy đủ có không quá một dấu “.”, nếu có dấu chấm thì sau đó có ít nhất một ký tự.

Không phân biệt chữ hoa và chữ thường. Trong mỗi thư mục không có các files hay thư mục con trùng tên.

Kết quả: Đưa ra file văn bản **CONTESTS.OUT** một số nguyên – số lượng kỳ thi được tải về.

Ví dụ:

CONTESTS.INP
22
C:\olymp\roi2005\aplusb\tests\01
C:\olymp\roi2005\aplusb\tests\01.a
C:\olymp\roi2005\aplusb\tests\02
C:\olymp\roi2005\aplusb\tests\02.a
C:\olymp\roi2005\aplusb\check.exe
C:\olymp\roi2005\gcd\tests\01
C:\olymp\roi2005\gcd\tests\01.a
C:\olymp\roi2005\gcd\tests\02
C:\olymp\roi2005\gcd\tests\02.a
C:\olymp\roi2005\gcd\check.cpp
C:\olymp\roi2005\gcd\solution.exe
C:\olymp\roi2006\aplusb\tests\01
C:\olymp\roi2006\aplusb\tests\01.a
C:\olymp\roi2006\aplusb\tests\03
C:\olymp\roi2006\aplusb\tests\03.a
C:\olymp\roi2006\aplusb\check.exe
C:\olymp\roi2006\gcd\tests\01
C:\olymp\roi2006\gcd\tests\01.a
C:\olymp\roi2006\gcd\tests\03
C:\olymp\roi2006\gcd\tests\02.a
C:\olymp\roi2006\gcd\check.cpp
C:\olymp\roi2006\gcd\solution.exe

CONTESTS.OUT
1

PHO MÁT

Trong kho của nhà máy sản xuất pho mát ở Flatland có một đàn chuột m con sinh sống. Chúng rất thích pho mát và nhiều khi phá hỏng cả lô hàng của nhà máy. Từ các thí nghiệm khoa học chúng ta đã biết rằng chuột là động vật rất thông minh. Một lần đàn chuột biết được kế hoạch của một phân xưởng. Sẽ có n bánh pho mát được sản xuất, bánh thứ i trở thành thành phẩm vào giờ r_i , có khối lượng là p_i và sẽ bị hỏng bắt đầu từ đầu giờ d_i .

Đàn chuột muốn ăn hết lượng pho mát của đợt hàng này. Dĩ nhiên, chúng chỉ ăn pho mát thành phẩm. Con chuột thứ j ăn được s_j gam trong một giờ. Chuột cũng là giống tham lam và hung dữ. Ở mỗi thời điểm chuột có thể chọn một bánh pho mát nào đó để ăn và chỉ có một con ăn bánh pho mát đó. Nó có thể bỏ dở bánh pho mát đang ăn để chuyển sang ăn bánh khác. Khi đó con chuột khác có thể tới ăn bánh pho mát bị bỏ dở. Ăn pho mát khi chưa bị hỏng vẫn ngon hơn. Tuy vậy chúng không muốn bỏ thừa ngay cả khi pho mát bị hỏng.

Gọi t là tổng thời gian ăn pho mát bị hỏng. Đàn chuột muốn lên kế hoạch ăn để t là nhỏ nhất. Ví dụ, đàn chuột có 2 con, phân xưởng sẽ sản xuất 2 bánh pho mát, bánh thứ nhất có khối lượng 13g, hoàn thành lúc 0 giờ và sẽ bị hỏng từ giờ thứ 4 trở đi, bánh pho mát thứ 2 có khối lượng 10g, hoàn thành lúc 1 giờ và hỏng từ giờ thứ 3 trở đi. Chú chuột thứ nhất ăn được 4g/giờ, chú chuột thứ 2 – 2g/giờ. Chú chuột thứ nhất sẽ ăn bánh pho mát thứ nhất một giờ sau đó chuyển sang bánh thứ 2 (bánh thứ nhất còn lại 9g), chú chuột thứ 2 ăn tiếp bánh pho mát thứ nhất. Sau 2.5 giờ chú chuột thứ nhất quay lại bánh pho mát đầu tiên (lúc này chỉ còn 4g) và ăn hết trong vòng một giờ, như vậy có 0.5 giờ chuột phải ăn pho mát đã bị hỏng.

Yêu cầu: Cho các số nguyên m, n, p_i, r_i, d_i, s_j ($1 \leq n, m \leq 30$, $1 \leq p_i \leq 10^5$, $0 \leq r_i < d_i \leq 10^7$, $1 \leq s_j \leq 10^5$). Hãy xác định thời gian t tối thiểu.

Dữ liệu: Vào từ file văn bản **CHEESE.INP**:

- Dòng đầu tiên chứa 2 số nguyên n và m ,
- Dòng thứ i trong n dòng tiếp theo chứa 3 số nguyên p_i , r_i và d_i ,
- Dòng thứ j trong m dòng còn lại chứa số nguyên s_j .

Kết quả: Đưa ra file văn bản **CHEESE.OUT** một số thực t với độ chính xác 4 chữ số sau dấu chấm thập phân.

Ví dụ:

CHEESE.INP			
2	2		
13	0	4	
10	1	3	
4			
2			

CHEESE.OUT
0.5000