

# Bài tập Greedy – Tuần 3

## Bài 1 – Mật độ Giao thông

Để nắm tình hình giao thông trên đường cao tốc mới xây dựng người ta đã tiến hành đo đạc thống kê khoảng cách trung bình giữa các phương tiện tham gia giao thông trên toàn tuyến vào giờ cao điểm và nhận được dãy số nguyên  $d_1, d_2, \dots, d_n$ , trong đó  $d_i$  là khoảng cách trung bình giữa các phương tiện giao thông trên đoạn đường thứ  $i$ . Hai đoạn đường  $i$  và  $j$  có tình trạng giao thông giống nhau bao nhiêu thì độ lệch  $h = |d_i - d_j|$  càng nhỏ bấy nhiêu. Hãy tính độ lệch của hai đoạn đường có tình trạng giao thông giống nhau nhất.

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

- Dòng đầu tiên chứa số nguyên  $n$  ( $1 < n \leq 10^6$ )
- Dòng thứ 2 chứa  $n$  số nguyên  $d_1, d_2, \dots, d_n$  ( $1 \leq d_i \leq 10^9, i = 1 \div n$ ). Các số trên một dòng ghi cách nhau một dấu cách.

**Kết quả:** Đưa ra file văn bản HIGHWAY.OUT một số nguyên – độ lệch tìm được.

**Ví dụ:**

HIGHWAY.INP	HIGHWAY.OUT
6 12 4 6 9 7 14	1

## Bài 2 – Shopaholic (Chứng nghiện mua sắm)

Lindsay là một tín đồ mua sắm. Bất cứ khi nào có giảm giá loại mua ba món tính tiền hai món, cô sẽ cảm thấy cần mua tất cả món hàng một cách điên dại. Bạn đã bó tay trước bệnh này của cô ấy, nhưng phải cố làm sao giúp hạn chế thiệt hại đối với túi tiền.

Bạn nhận thấy rằng các cửa hàng đưa ra khuyến mãi này thường chọn lọc món hàng được khuyến mãi, đó luôn là những món rẻ nhất. Ví dụ khi bạn của bạn đến quầy thanh toán với bảy món hàng trị giá 400, 350, 300, 250, 200, 150 và 100 đồng, cô ấy phải trả 1500 đồng. Như vậy cô ấy được giảm 250 đồng. Bạn thấy rằng

nếu cô ấy thanh toán 3 lần, cô ấy có thể được giảm giá nhiều hơn. Ví dụ nếu thanh toán 3 món trị giá 400, 300 và 250 đồng, cô ấy sẽ được giảm 250 đồng ở lần đầu. Lần kế tiếp cô ấy đem món hàng trị giá 150 đồng sẽ không được giảm giá nhưng lần thứ ba thanh toán những món cuối trị giá 350, 200 và 100 cô ấy sẽ được giảm thêm 100 đồng, tổng cộng được giảm 350 đồng.

Việc của bạn là tìm số tiền giảm giá tối đa mà Lindsay có thể nhận được.

**Dữ liệu:** Vào từ file SHOP.INP

Dòng đầu cho biết số trường hợp,  $1 \leq t \leq 20$ . Mỗi trường hợp chứa hai dòng dữ liệu. Dòng đầu cho biết số món hàng Lindsay đang mua,  $1 \leq n \leq 20000$ . Dòng kế tiếp cho biết giá của những món này,  $1 \leq p_i \leq 20000$ .

**Kết quả:** Xuất ra file SHOP.OUT. Với mỗi trường hợp, xuất ra 1 số trên 1 dòng cho biết giá trị giảm tối đa Lindsay có thể nhận bằng cách chọn các món hàng cô ấy sẽ thanh toán cùng một lúc.

**Ví dụ:**

SHOP.INP	SHOP.OUT
1	400
6	
400 100 200 350 300 250	

### Bài 3 – Giảm tải công việc

Bạn hiện có **N** công việc và sếp của bạn yêu cầu đến cuối ngày bạn chỉ còn lại **M** công việc. Hy vọng duy nhất là thuê các công ty dịch vụ bên ngoài giúp đỡ. Có nhiều công ty khác nhau đưa ra các gói hỗ trợ như sau:

Với số tiền **A** họ sẽ hỗ trợ giải quyết một công việc.

Với số tiền **B** họ sẽ giải quyết một nửa số công việc (công việc còn lại làm tròn xuống khi cần thiết).

Lưu ý rằng công việc không thể nào được giảm xuống nhỏ hơn **M**.

Nhiệm vụ bây giờ của bạn là tạo ra một bảng sắp xếp tên các công ty và chi phí tối thiểu tương ứng để giải quyết vấn đề công việc.

**Dữ liệu:** Vào từ file REDUCTION.INP.

Dòng đầu tiên chứa một số dương cho biết số bộ dữ liệu. Mỗi bộ dữ liệu bắt đầu với ba số nguyên dương cách nhau bởi khoảng trắng: **N** – khối lượng công việc ban đầu, **M** – khối lượng công việc mục tiêu và **L** – số công ty có thể cung cấp dịch vụ cho bạn ( $1 \leq M \leq N \leq 100000$ ,  $1 \leq L \leq 100$ ). **L** dòng tiếp theo có định dạng “[*tên công ty*]:**A,B**”, trong đó **A** và **B** là mức giá như mô tả ở trên của mỗi công ty. ( $0 \leq A, B \leq 10000$ ). Chiều dài tên công ty dịch vụ từ 1 đến 16 ký tự và chỉ bao gồm các ký tự in hoa. Tên các công ty là duy nhất.

**Kết quả:** Xuất ra file REDUCTION.OUT.

Với mỗi bộ dữ liệu, in ra “Case X”, với X là số thứ tự của bộ dữ liệu, trên một dòng, sau đó là bảng bao gồm tên công ty và chi phí tối thiểu tương ứng, sắp xếp theo chi phí tối thiểu tăng dần. Các công ty có chi phí tối thiểu giống nhau sẽ được sắp xếp theo tên. Với mỗi dòng trong bảng, in ra tên công ty, khoảng trắng, chi phí tối thiểu cần để giải quyết vấn đề.

**Ví dụ:**

REDUCTION.INP	REDUCTION.OUT
2	Case 1
100 5 3	C 7
A:1,10	B 22
B:2,5	A 37
C:3,1	Case 2
1123 1122 5	E 0
B:50,300	A 1
A:1,1000	D 1
C:10,10	C 10
D:1,50	B 50
E:0,0	

## Bài 4 – Giao dịch rượu

Gergovia có một con đường và mỗi cư dân trong thành phố là một nhà bán rượu. Nền kinh tế này hoạt động như thế nào? Khá đơn giản: mỗi người mua rượu từ một cư dân khác của thành phố. Mỗi ngày mỗi cư dân quyết định họ sẽ mua hoặc bán bao nhiêu rượu. Thật thú vị là cung và cầu luôn bằng nhau nên mỗi cư dân đều nhận được thứ họ muốn.

Tuy nhiên có một vấn đề: Vận chuyển rượu từ nhà này sang nhà khác đòi hỏi phải bỏ công lao động. Bởi vì tất cả rượu đều ngon như nhau, các cư dân của Gergovia không quan tâm người mà mình đang giao dịch là ai, họ chỉ quan tâm đang bán hay mua một lượng rượu cụ thể là bao nhiêu. Họ đủ sáng suốt để tìm ra cách giao dịch sao cho tổng khối lượng công việc vận chuyển là ít nhất.

Trong bài toán này bạn cần cấu trúc lại các giao dịch trong một ngày ở Gergovia. Để đơn giản chúng ta giả sử rằng các ngôi nhà được xây dựng dọc theo một đường thẳng với khoảng cách bằng nhau giữa các nhà lân cận. Vận chuyển một chai rượu từ một nhà này đến nhà kế bên tốn 1 đơn vị lao động.

**Dữ liệu:** Vào từ file TRADING.INP, gồm nhiều bộ dữ liệu. Mỗi bộ dữ liệu bắt đầu với số lượng dân cư  $n$  ( $2 \leq n \leq 100000$ ). Các dòng tiếp theo chứa  $n$  số nguyên  $a_i$  ( $-1000 \leq a_i \leq 1000$ ). Nếu  $a_i \geq 0$ , điều đó có nghĩa là cư dân sống trong nhà thứ  $i$  muốn mua  $a_i$  chai rượu, ngược lại nếu  $a_i < 0$ , người đó muốn bán  $-a_i$  chai rượu. Tổng  $a_i$  bằng 0. Sau bộ test cuối cùng là một dòng chứa số 0.

**Kết quả:** Xuất ra file TRADING.OUT. Với mỗi bộ dữ liệu in ra số đơn vị lao động tối thiểu cần phải vận chuyển để mỗi cư dân thỏa mãn yêu cầu mua bán của mình.

**Ví dụ:**

TRADING.INP	TRADING.OUT
5	9
5 -4 1 -3 1	9000
6	
-1000 -1000 -1000 1000 1000 1000	
0	