



# Ngôn ngữ lập trình C++

## BÀI TẬP NÂNG CAO



### 1. Tính tổng – CalSum.Cpp

Cho dãy số nguyên  $A_1, A_2, \dots, A_N$  và số  $X$ . Tính tổng  $S = |A_1 - X| + |A_2 - X| + \dots + |A_N - X|$ .

**Dữ liệu** cho trong file CalSum.Inp gồm:

- Dòng đầu ghi số nguyên dương  $N$  ( $1 \leq N \leq 100$ ).
- Dòng thứ 2 ghi  $N$  số nguyên  $A_1, A_2, \dots, A_N$  ( $0 \leq A_i \leq 10^9$ ).
- Dòng thứ 3 ghi số nguyên  $X$  ( $0 \leq X \leq 10^9$ ).

**Kết quả** ghi ra file CalSum.Out là giá trị của  $S$ .

*Ví dụ:*

CalSum.Inp	CalSum.Out
3 1 2 3 2	2



### 2. Dịch sỏi trên trục số - MoveStone.Cpp

Có  $N$  viên sỏi đặt trên trục số, viên sỏi thứ  $i$  đặt tại vị trí tọa độ  $x_i$ . Mỗi lần dịch chuyển, chỉ có thể dịch chuyển viên sỏi sang tọa độ kề trái hoặc kề phải, tức là nếu viên sỏi đang ở vị trí tọa độ  $x$  thì mỗi lần dịch chuyển, ta có thể dịch sang tọa độ  $x + 1$  hoặc  $x - 1$ .


**Yêu cầu:** Tìm cách dịch chuyển các viên sỏi với số lần dịch chuyển ít nhất để  $N$  viên sỏi được dịch đến một vị trí tọa độ.

**Dữ liệu** cho trong file MoveStone.Inp gồm:

- Dòng đầu ghi số nguyên dương  $N$ .
- Dòng thứ 2 ghi  $N$  tọa độ nguyên  $x_1, x_2, \dots, x_N$ .

**Kết quả** ghi ra file MoveStone.Out là số lần dịch chuyển ít nhất để đưa  $N$  viên sỏi về một vị trí tọa độ.

*Ví dụ:*

MoveStone.Inp	MoveStone.Out	Hình vẽ và giải thích
3 1 2 3	2	 Lần 1: Dịch viên sỏi ở vị trí 1 sang vị trí 2. Lần 2: Dịch viên sỏi ở vị trí 3 sang vị trí 2.

**Giới hạn:**

- 50% số test ứng với  $N \leq 1000; 0 \leq x_i \leq 1000$ .
- 50% số test ứng với  $N \leq 1000000; 0 \leq x_i \leq 1.000.000.000$ .

**Một trung tâm**

Dọc con đường thẳng tấp của đất nước XYX có  $N$  cửa hàng của tập đoàn YYX. Các cửa hàng có vị trí tại các tọa độ  $x_1, x_2, \dots, x_N$ . Tập đoàn muốn xây dựng 1 trung tâm tại vị trí  $x_i$  để cung ứng sản phẩm cho các cửa hàng. Mỗi cửa hàng  $x_h$  chi phí đi lại là  $|x_i - x_h|$ . Tổng chi phí đi lại của  $N$  cửa hàng đến trung tâm là:  $S = \sum_{h=1}^N |x_i - x_h|$ .

**Yêu cầu:** Tìm giá trị nhỏ nhất của  $S$  có thể đạt được.

**Dữ liệu** cho trong file OneC.Inp gồm:

- Dòng đầu ghi số nguyên dương  $N$  là số cửa hàng.
- Dòng thứ 2 ghi  $N$  số nguyên  $x_1, x_2, \dots, x_N$  ( $0 \leq x_1 < x_2 < \dots < x_N \leq 10^7$ ).

**Kết quả** ghi ra file OneC.Out là tổng chi phí đi lại nhỏ nhất của  $N$  cửa hàng.

Ví dụ:

OneC.Inp	OneC.Out
4 1 2 3 7	7

**Giới hạn:**

- Sub1:  $N \leq 1000$ ;
- Sub2:  $N \leq 200000$ ;

**Hai trung tâm**

Dọc con đường thẳng tấp của đất nước XYX có  $N$  cửa hàng của tập đoàn YYX. Các cửa hàng có vị trí tại các tọa độ  $x_1, x_2, \dots, x_N$ . Tập đoàn muốn xây dựng 2 trung tâm tại hai vị trí  $x_i$  và  $x_j$  để cung ứng sản phẩm cho các cửa hàng. Mỗi cửa hàng  $x_h$  chỉ chọn một trung tâm  $x_t$  ( $t = i$  hoặc  $t = j$ ) để lấy sản phẩm, khi đó chi phí đi lại là  $|x_t - x_h|$ .

**Yêu cầu:** tìm hai vị trí  $i$  và  $j$  để tổng chi phí đi lại của  $N$  cửa hàng là nhỏ nhất.

**Dữ liệu** cho trong file TwoC.Inp gồm:

- Dòng đầu ghi số nguyên dương  $N$  là số cửa hàng.
- Dòng thứ 2 ghi  $N$  số nguyên  $x_1, x_2, \dots, x_N$  ( $0 \leq x_1 < x_2 < \dots < x_N \leq 10^7$ ).

**Kết quả** ghi ra file TwoC.Out là tổng chi phí đi lại nhỏ nhất của  $N$  cửa hàng.

Ví dụ:

TwoC.Inp	TwoC.Out
4 1 2 3 7	2

**Giới hạn:**

- Sub1:  $N \leq 1000$ ;
- Sub2:  $N \leq 200000$ ;

**5☀. Vận chuyển đồ cổ**

Các nhà khảo cổ vừa tìm được  $N$  đồ vật quý hiếm. Các đồ vật này được định vị trên một đường thẳng. Đồ vật thứ  $i$  định vị tại vị trí  $X_i$  ( $i = 1, 2, \dots, N$ ), trong đó  $0 \leq X_1 < X_2 < \dots < X_N \leq 10^8$  và có khối lượng là  $A_i$ . Do các đồ vật cổ vừa quý hiếm nhưng rất nặng nên các nhà khảo cổ quyết định sẽ xây dựng một ngôi nhà tạm tại vị trí tọa độ  $U$  trên đường thẳng để chứa  $N$  đồ vật. Để chuyển đồ vật  $i$  đến ngôi nhà mất chi phí là  $A_i \times |U - X_i|$ . Như vậy, tổng chi phí để vận chuyển  $N$  đồ cổ đến ngôi nhà là:

$$S = \sum_{i=1}^N A_i \times |U - X_i|$$

**Yêu cầu:** Tính giá trị nhỏ nhất của  $S$ .

**Dữ liệu** cho trong file **Antique.Inp** gồm:

- Dòng thứ 1 ghi số nguyên dương  $N$  là số đồ vật.
- Dòng thứ 2 ghi  $N$  số nguyên  $X_1, X_2, \dots, X_N$  là tọa độ của  $N$  đồ vật.
- Dòng thứ 3 ghi  $N$  số nguyên  $A_1, A_2, \dots, A_N$  là khối lượng của  $N$  đồ vật ( $0 < A_i \leq 10^3$ ).

**Kết quả** ghi ra file **Antique.Out** là giá trị nhỏ nhất của  $S$  đạt được.

Ví dụ:

Antique.Inp	Antique.Out
3	8
1 4 6	
10 1 1	

**Giới hạn:**

- Sub1:  $X_N \leq 10^3; N \leq 10^3$ ;
- Sub2:  $N \leq 10^5$ ;

**6☀. KHO LÚA (IOI - 2011)**

Quê tôi có một con đường dài và thẳng gọi là con đường Lúa. Dọc theo con đường này có  $R$  cánh đồng được đánh số từ 0 đến  $R - 1$ . Mỗi cánh đồng được định vị ở tọa độ nguyên trong khoảng từ 1 đến  $L$ . Các đồng được đánh số theo thứ tự không giảm của tọa độ. Một cách hình thức chính xác, với  $0 \leq i < R$ , cánh đồng  $i$  ở tại tọa độ  $X[i]$  ( $1 \leq X[0] \leq X[1] \leq \dots \leq X[R - 1] \leq L$ ). Chú ý rằng, có thể có nhiều cánh đồng có cùng một tọa độ.

Chúng tôi cần xây dựng một kho lúa chung chứa được nhiều lúa thu hoạch trong vụ mùa thì càng tốt. Giống như các cánh đồng, kho lúa phải được đặt ở tọa độ nguyên trong phạm vi từ 1 tới  $L$ . Kho lúa có thể đặt ở vị trí tùy ý, kể cả ở vị trí đã có một hoặc nhiều hơn một cánh đồng.

Vào mùa thu hoạch, mỗi cánh đồng sản xuất lượng lúa chất đầy đúng 1 xe tải. Để chuyển lúa về kho, thành phố phải thuê một người lái xe tải. Người lái xe đòi thù lao 1 Baht để lái một xe lúa đi một đơn vị độ dài trên đường đi từ cánh đồng về kho. Nói cách khác, giá vận chuyển lúa từ một cánh đồng về kho có giá trị số đúng bằng chênh lệch giữa hai tọa độ của chúng.

Đáng tiếc do ngân sách dành cho vụ mùa này rất hạn hẹp: Chúng tôi chỉ được chi tối đa **B** Baht cho việc vận chuyển. Nhiệm vụ của bạn là tìm giúp chúng tôi vị trí đặt kho lúa chứa được nhiều lúa nhất.



**Dữ liệu** cho trong file RICE.INP như sau:

- Dòng đầu ghi ba số nguyên  $R, L, B$ .
- $R$  dòng tiếp theo ghi  $R$  số nguyên là tọa độ của các cánh đồng ( $X[0], X[1], \dots, X[R-1]$ ).

**Kết quả** ghi ra file RICE.OUT là lượng lúa lớn nhất mà xe có thể chuyển tới.

*Giới hạn:*

- 17% số test ứng với  $1 \leq R \leq 100; 1 \leq L \leq 100; 0 \leq B \leq 10.000$
- 25% số test ứng với  $1 \leq R \leq 500; 1 \leq L \leq 10.000; 0 \leq B \leq 10^6$
- 26% số test ứng với  $1 \leq R \leq 5.000, 1 \leq L \leq 10^6; 0 \leq B \leq 2 \cdot 10^9$
- 32% số test ứng với  $1 \leq R \leq 10^5; 1 \leq L \leq 10^9, 0 \leq B \leq 2 \cdot 10^{15}$

*Ví dụ:*

RICE.INP	RICE.OUT
5 20 6 1 2 10 12 14	3

