



Ngôn ngữ lập trình C++

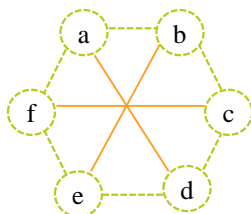
BÀI TẬP LẬP TRÌNH CƠ BẢN VÀ NÂNG CAO

A. Phần cơ bản



1 ☀

Chọn số trên đỉnh lục giác – HexaGonNum.Cpp



Cho hình lục giác có 6 số được ghi tại 6 đỉnh của lục giác (hình vẽ).

Hãy chọn hai số ở vị trí hai đỉnh đối diện sao cho có tổng lớn nhất.

Dữ liệu: Nhập 6 số a, b, c, d, e, f .

Kết quả: Đưa ra tổng của 2 số ở vị trí 2 đỉnh đối xứng có giá trị lớn nhất.

Ví dụ:

Input	Output	Hình minh họa
1 2 3 4 5 9	12	



2 ☀

. Robot nhảy trên đường thẳng – JumpInLine.Cpp

Có một đường thẳng được chia vạch khoảng cách. Vạch 0, vạch 1, vạch 2,



0 1 2 3 4 5 6 7 8 9 10

Có một con robot đang đứng tại vạch 0. Robot sẽ nhảy N bước, bước nhảy thứ i , robot nhảy thêm được i vạch ($i = 1, 2, 3, \dots, N$).

Yêu cầu: Sau N bước nhảy, robot sẽ đến vạch nào trên đường thẳng.

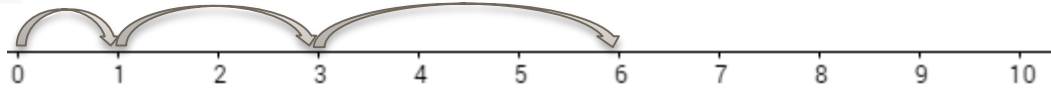
Dữ liệu nhập số nguyên dương N ($N \leq 1000$).

Kết quả đưa ra vạch mà robot sẽ đến sau N bước nhảy.

Ví dụ:

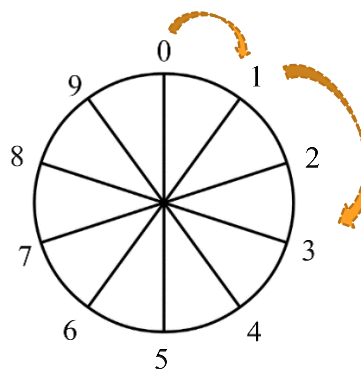


Input	Output
3	6



3 Robot nhảy trên đường tròn – JumpInCircle.Cpp

Cho đường tròn có chia thành $M + 1$ vạch, bắt đầu từ vạch 0, kết thúc từ vạch M theo chiều kim đồng hồ (hình vẽ với $M = 9$).



Robot sẽ nhảy N bước, bước nhảy thứ i , robot nhảy thêm được i vạch ($i = 1, 2, 3, \dots, N$) theo chiều kim đồng hồ.

Yêu cầu: Sau N bước nhảy, robot sẽ đến vạch nào trên đường tròn.

Dữ liệu nhập hai số nguyên dương N và M ($N \leq 1000$, $M \leq 1000000$).

Kết quả đưa ra vạch mà robot sẽ đến sau N bước nhảy.

Ví dụ:

Input	Output	Giải thích
4 9	0	Lần nhảy 1: đến vạch 1 Lần nhảy 2: đến vạch 3 Lần nhảy 3: đến vạch 6 Lần nhảy 4: đến vạch 0

**4☀. Chọn ba số - Select3Num.Cpp**

Cho dãy số nguyên A_1, A_2, \dots, A_N . Hãy chọn 3 số hạng không ở vị trí kề nhau trong dãy sao cho tổng 3 số được chọn là lớn nhất.

Dữ liệu: Nhập N ($6 \leq N \leq 100$).

Nhập N số nguyên A_1, A_2, \dots, A_N ($|A_i| \leq 10000$).

Kết quả: Đưa ra tổng 3 số chọn được.

Ví dụ:

Input	Output	Giải thích
6 9 8 7 0 0 4	20	Chọn 3 số hạng có giá trị: 9, 7, 4.

A. Phân Nâng cao**1☀. Chọn ô trên đường chéo (1)**

Cho bảng số gồm N dòng và M cột. Các dòng được đánh số từ 1 đến N (từ trên xuống dưới), các cột được đánh số từ 1 đến M (từ trái sang phải). Ô ở dòng i cột j , ta kí hiệu là ô (i, j) và có ghi số nguyên A_{ij} .

Yêu cầu: Tìm ba ô nằm trên đường chéo: (i, j) ; $(i+1, j+1)$, $(i+2, j+2)$ sao cho tổng ba số ghi trên 3 ô này có giá trị lớn nhất.

Dữ liệu cho trong file **Diagon1.Inp** gồm:

- Dòng đầu ghi hai số nguyên dương N và M ($3 \leq N, M \leq 1000$).
- N dòng sau, mỗi dòng ghi M số nguyên thuộc tập $[-1000; 1000]$.

Kết quả ghi ra file **Diagon1.Out** là tổng lớn nhất của 3 ô tìm được.

Ví dụ:

Diagon1.Inp	Diagon1.Out
4 5 1 2 3 4 5 1 1 1 1 1 9 9 9 1 1 0 0 0 0 0	11

**2☀. Chọn ô trên đường chéo (2)**

Cho bảng số gồm N dòng và M cột. Các dòng được đánh số từ 1 đến N (từ trên xuống dưới), các cột được đánh số từ 1 đến M (từ trái sang phải). Ô ở dòng i cột j , ta kí hiệu là ô (i, j) và có ghi số nguyên A_{ij} .

Yêu cầu: Tìm ba ô nằm trên đường chéo: (i, j) , $(i+u, j+u)$, $(i+v, j+v)$ sao cho tổng ba số ghi trên 3 ô này có giá trị lớn nhất ($1 \leq u < v$).



Dữ liệu cho trong file **Diagon2.Inp** gồm:

- Dòng đầu ghi hai số nguyên dương N và M ($3 \leq N, M \leq 1000$).
- N dòng sau, mỗi dòng ghi M số nguyên thuộc tập $[-1000; 1000]$.

Kết quả ghi ra file **Diagon2.Out** là tổng lớn nhất của 3 ô tìm được.

Ví dụ:

Diagon2.Inp	Diagon2.Out
4 5	11
1 2 3 4 5	
1 0 1 1 1	
9 9 9 1 1	
0 0 0 1 0	



3. Chọn ô trên đường chéo (3)

Cho bảng số gồm N dòng và M cột. Các dòng được đánh số từ 1 đến N (từ trên xuống dưới), các cột được đánh số từ 1 đến M (từ trái sang phải). Ô ở dòng i cột j , ta kí hiệu là ô (i, j) và có ghi số nguyên A_{ij} .

Yêu cầu: Tìm 6 ô nằm trên đường chéo:

$$(i, j), (i + 1, j + 1), (i + 2, j + 2);$$

$$(u, v), (u + 1, v - 1), (u + 2, v - 2) \text{ sao cho tổng 6 số ghi trên 6 ô này có giá trị lớn nhất.}$$

Dữ liệu cho trong file **Diagon3.Inp** gồm:

- Dòng đầu ghi hai số nguyên dương N và M ($3 \leq N, M \leq 1000$).
- N dòng sau, mỗi dòng ghi M số nguyên thuộc tập $[-1000; 1000]$.

Kết quả ghi ra file **Diagon3.Out** là tổng lớn nhất của 6 ô tìm được (chú ý là 6 ô phân biệt).

Ví dụ:

Diagon3.Inp	Diagon3.Out
4 5	25
1 2 3 4 5	
1 1 1 1 1	
9 9 9 1 1	
0 0 0 0 0	



4. Bộ ba chênh lệch

Cho dãy số nguyên A gồm N số nguyên A_1, A_2, \dots, A_N . Tìm số các bộ 3 chỉ số (i, j, k) sao cho:

- $1 \leq i < j < k \leq N$;
- $\max(A_i, A_j, A_k) - \min(A_i, A_j, A_k) \leq 2$.

Dữ liệu cho trong file **Tuple3.Inp** gồm:

- Dòng đầu ghi số nguyên dương N .



- Dòng thứ 2 ghi N số nguyên A_1, A_2, \dots, A_N .

Kết quả ghi ra file **Tuple3.Out** là số các chỉ số i thỏa mãn điều kiện trên.

Ví dụ:

Tuple3.Inp	Tuple3.Out	Giải thích
5 3 6 7 5 9	1	Chọn $A[2], A[3], A[4]$

Giới hạn:

- $1 \leq A_i \leq 10^6$;
- Có 50% số test ứng với $N \leq 1000$;
- Có 50% số test ứng với $N \leq 2 \times 10^5$.