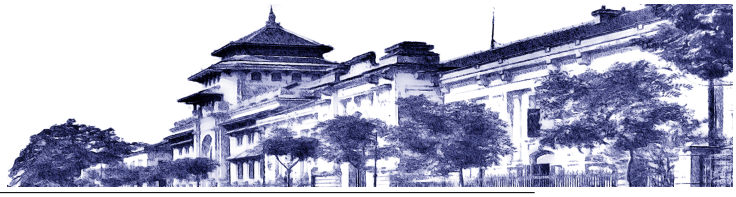
ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
Khoa Vật lý, Bộ môn Tin học Vật lý
P. 308D-T1, 334 Nguyễn Trãi, Thanh Xuân, Hà Nội



Phần 3

Nội dung thực hành

– Vận dụng các cấu trúc điều khiển
– Làm việc với mảng số 1 chiều: *khai báo, xuất/nhập, thao tác với các phần tử trong mảng*

Ví dụ

1. Nhập dữ liệu cho mảng 1 chiều

```c
#include <stdio.h>

main() {
    int a[50], i, n;            // Khai bao mang a chua toi da 50 phan tu

    puts("Nhap n: ");
    scanf("%d", &n);            // Nhap so phan tu cua mang

    for(i = 0; i < n; i++) {
        printf("Phan tu thu %d = ", i);
        scanf("%d", &a[i]);     // Nhap tung phan tu tu ban phim
    }

    for(i=0;i< n;i++)
        printf("%d\t", a[i]);   // In ra cac phan tu cua mang
}
```

2. Trung bình cộng của n số nguyên

```c
#include <stdio.h>

main() {
    int i, n, sum = 0, a[20];   // Khai bao mang a chua toi da 20 phan tu

    printf("Nhap vao gia tri n: ");
    scanf("%d", &n);            // Nhap n (n <= 20)

    for(i = 0; i < n; i++) {
        printf("Nhap vao phan tu thu %d: ", i + 1);
        scanf("%d", &a[i]);     // Nhap tung phan tu tu ban phim
        sum = sum + a[i];       // Tinh tong cac phan tu
    }

    printf("Trung binh cong: %f \n", (float)sum/n); // Tinh trung binh cong va in ra
}
```

Chú ý:
Nếu mảng có N phần tử, chỉ số của phần tử đầu tiên là 0, chỉ số của phần tử cuối cùng là $N-1$.

Bài tập

1. Cho một mảng có n phần tử. Tìm giá trị nhỏ nhất và lớn nhất của mảng đó.
Input
dòng thứ nhất là số nguyên n
dòng thứ hai là n số nguyên
Output
2 số nguyên là giá trị nhỏ nhất và lớn nhất của mảng

Input	Output
7 5 3 6 2 7 1 4	1 7

2. Dãy Fibonacci là dãy vô hạn các số tự nhiên bắt đầu bằng hai phần tử 0 và 1, được định nghĩa như sau:

$$F_n = \begin{cases} 0 & \text{nếu } n = 0 \\ 1 & \text{nếu } n = 1 \\ F_{n-1} + F_{n-2} & \text{nếu } n > 1 \end{cases}$$

Tính phần tử F_n của dãy Fibonacci.

Input
1 số nguyên n
Output
1 số nguyên là giá trị của F_n

Input	Output
8	21

👉 Mở rộng bài toán: Không sử dụng mảng để tính và in ra F_n.

3. Cho một mảng có n phần tử. Lọc ra các số dương (≥ 0) và các số âm (< 0) để lưu vào hai mảng khác nhau.

Input
dòng thứ nhất là số nguyên n
dòng thứ hai là n số nguyên
Output
dòng thứ nhất là các số trong mảng dương
dòng thứ hai là các số trong mảng âm

Input	Output
8 5 -3 6 0 -2 -7 1 4	5 6 0 1 4 -3 -2 -7

4. Cho một mảng có n phần tử. Chèn thêm một phần tử vào một vị trí được chọn trong mảng.

Input
dòng thứ nhất là số nguyên n
dòng thứ hai là n số nguyên
dòng thứ ba là 2 số: vị trí và giá trị phần tử chèn thêm
Output
$n + 1$ số là các phần tử của mảng sau khi đã chèn thêm

Input	Output
7 5 3 6 2 7 1 4 5 0	5 3 6 2 0 7 1 4

👉 Mở rộng bài toán: Tương tự, thực hiện thao tác xóa một phần tử của mảng.

5. Cho một mảng có n phần tử. Đảo ngược mảng đã cho.

Input
dòng thứ nhất là số nguyên n
dòng thứ hai là n số nguyên
Output
n số là các phần tử của mảng sau khi đã đảo ngược

Input	Output
7 5 3 6 2 7 1 4	4 1 7 2 6 3 5

👉 Gợi ý: Sử dụng thao tác đổi giá trị của hai biến cho nhau (*swap*).

6. Cho một mảng có n phần tử. Sắp xếp lại các phần tử của mảng đó theo thứ tự tăng dần.

Input
dòng thứ nhất là số nguyên n
dòng thứ hai là n số nguyên
Output
n số là các phần tử của mảng sau khi đã sắp xếp

Input	Output
7 5 3 6 2 7 1 4	1 2 3 4 5 6 7

👉 Mở rộng bài toán:
+ Bên cạnh các thuật toán có độ phức tạp $O(n^2)$ như sắp xếp chèn (*insertion sort*) hay sắp xếp nổi bọt (*bubble sort*), hãy viết một chương trình sử dụng một thuật toán nào đó có độ phức tạp $O(n \log n)$.
+ Tham khảo thêm các thuật toán sắp xếp tại *link*.
+ Áp dụng: Cho một mảng, hãy xem xét sự trùng nhau về mặt giá trị của các phần tử để chỉ giữ lại 1 phần tử và loại bỏ các phần tử trùng lặp ra khỏi mảng.

7. Cho một mảng có n phần tử. Tìm độ dài đoạn con dài nhất của mảng bao gồm các phần tử bằng nhau.

Input
dòng thứ nhất là số nguyên n
dòng thứ hai là n số nguyên
Output
3 số nguyên: độ dài đoạn con, vị trí bắt đầu và kết thúc của đoạn

Input	Output
10 1 7 2 2 4 1 1 1 7 7	3 6 8

8. Cho một mảng có n phần tử. Cho m truy vấn có dạng (i, j), hãy tính tổng các phần tử của mảng từ phần tử thứ i đến phần tử thứ j.

Input
dòng thứ nhất là 2 số nguyên n và m
dòng thứ hai là n số nguyên
tiếp đến là m dòng, mỗi dòng chứa 2 số i và j
Output
m dòng, mỗi dòng là 1 số để trả lời truy vấn tương ứng

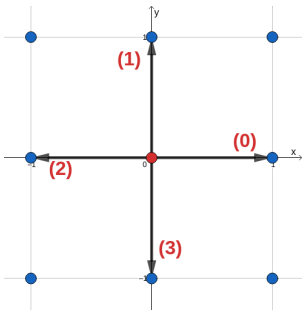
Input	Output
5 3 1 -8 -1 2 9 1 5 3 3 2 4	3 -1 -7

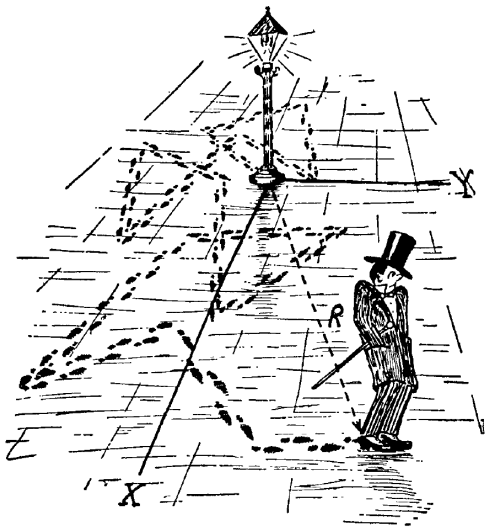
☞ Mở rộng bài toán: Tìm cách giải quyết bài toán với n và m khoảng 10^8.

9. Một người đàn ông say rượu đứng ở cột điện. Vì quá say nên anh ta không kiểm soát được bước đi của mình và bước đi các bước ngẫu nhiên về các phía.
Giả sử cột đèn ở vị trí tọa độ $(0, 0)$. Mỗi bước đi ngẫu nhiên có độ dài là 1, có thể theo 4 hướng đánh số từ 0 đến 3 (như hình vẽ). Hỏi sau một số n bước đi ngẫu nhiên thì anh ta có tọa độ bao nhiêu?

Input
dòng thứ nhất là số nguyên n
dòng thứ hai là n số nguyên có giá trị từ 0 đến 3
Output
2 số x và y là tọa độ anh chàng say rượu

Input	Output
10 1 2 0 0 1 2 3 0 1 3	1 1





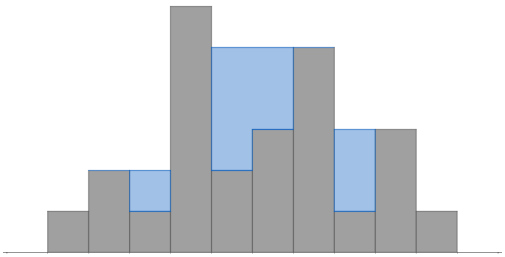
☞ Mở rộng bài toán:
+ Liệu sau một số n rất lớn thì anh ta có trở về cột đèn ban đầu? Nếu không thì khoảng cách trung bình của anh ta tới cột đèn là bao nhiêu?
+ Không dùng cấu trúc điều khiển *if* hay *switch* để viết chương trình.
+ Thay vì nhập số bước đi ngẫu nhiên bằng tay, có thể sử dụng hàm tạo số ngẫu nhiên *rand()* của thư viện *stdlib.h*.

☞ : Bài toán bước ngẫu nhiên (*random walk*) có tính ứng dụng cao trong nhiều lĩnh vực, từ toán học, vật lý, sinh học, cho đến tâm lý học và kinh tế học!

10. Một hệ thống mái che được xây dựng để che mưa. Mái che có bề rộng là n, được chia làm n phần có độ dài như nhau. Độ cao của mỗi phần là $h_1, h_2, \ldots, h_n$. Khi trời mưa, một phần nước sẽ đọng lại trên mái và một phần sẽ thoát ra ngoài theo hai bên trái và phải của mái che.
Nhằm mục đích bảo trì mái che, hãy viết chương trình tính lượng nước lớn nhất có thể đọng lại trên mái che.

Input
dòng thứ nhất là số nguyên n
dòng thứ hai là n số nguyên h_i
Output
số lượng nước đọng lại trên mái che

Input	Output
10 1 2 1 6 2 3 5 1 2 1	8



👉 Bài toán lấy ý tưởng từ bài: *V11WATER - Nước đọng* trên *VNOI*.

11. Cho một dãy số nguyên gồm n phần tử $a_1, a_2, \ldots, a_n$. Biết rằng dãy con tăng đơn điệu là một dãy $a_{i_1}, a_{i_2}, \ldots, a_{i_k}$ thỏa mãn $i_1 < i_2 < \cdots < i_k$ và $a_{i_1} < a_{i_2} < \cdots < a_{i_k}$. Hãy cho biết dãy con tăng đơn điệu dài nhất (*LIS - Longest Increasing Subsequence*) của dãy đã cho có bao nhiêu phần tử.

Input
dòng thứ nhất là số nguyên n
dòng thứ hai là n số nguyên a_i
Output
độ dài của dãy con tăng đơn điệu dài nhất

Input	Output
6 1 2 5 4 6 2	4

Giải thích test ví dụ: dãy con dài nhất là dãy $a_1 = 1 < a_2 = 2 < a_4 = 4 < a_5 = 6$.

👉 Bài toán lấy ý tưởng từ bài: *LIQ* và *LIS* trên *VNOI*.

12. Cho 2 dãy số nguyên a gồm m phần tử, b gồm n phần tử. Một dãy con của a là một dãy gồm một số các phần tử của a (không thay đổi thứ tự và không cần liên tiếp). Tương tự với b. Tìm độ dài dãy con chung dài nhất (*LCS - Longest common subsequence*) của hai dãy a và b.

Input
dòng thứ nhất là 2 số m và n
dòng thứ hai là m số nguyên a_i
dòng thứ ba là n số nguyên b_i
Output
độ dài của dãy con chung dài nhất của a và b

Input	Output
6 7 1 2 5 4 6 2 1 5 3 2 4 1 6	4

Giải thích test ví dụ: dãy con chung dài nhất có thể là $1, 2, 4, 6$ *hoặc* $1, 5, 4, 6$.

👉 Bài toán lấy ý tưởng từ bài: *VOSLIS* và *LCS2X* trên *VNOI*.