

1. ĐẾM SỐ PHẦN TỬ TRONG MẢNG – HISTOGRAM

Bài toán: Cho $a[n]$ là một mảng với phần gồm n phần tử. Ta cần đếm số lượng từng phần tử trong mảng.

Ví dụ:

Với $n = 8$ $a[8] = (5, 2, 8, 4, 5, 6, 7, 8)$

Số phần tử 2 là 1

Số phần tử 4 là 1

Số phần tử 5 là 2

Số phần tử 6 là 1

Số phần tử 7 là 1

Số phần tử 8 là 2

Giải pháp:

Phương pháp thông thường các bạn sẽ bắt nguồn từ bài toán đếm một phần tử x nào đó xuất hiện trong mảng bao nhiêu lần, sau đó dùng lại đoạn code đó để đếm tất cả các phần tử trong mảng.

```
4  main()
5  {
6      int Dem[10];
7
8      for ( x=0; x<10; x++)
9      {
10         Dem[x] = 0;
11         for ( i=0; i<n; i++)
12             if (a[i] == x)
13                 Dem[x]++;
14     }
15 }
```

Cách làm này hơi tốn kém...

Phương án đề xuất:

Sẽ tốt hơn nếu các bạn chỉ chạy một lượt qua mảng, và với mỗi giá trị, ta tăng biến đếm $a[i]$ lên một đơn vị. Ví dụ dưới, ta có thể dùng bản thân giá trị đó làm chỉ số cho mảng các biến đếm.

Sau đây là đoạn mã để duyệt một mảng các số nguyên và phát sinh ra số lượng các phần tử.

```

4  main()
5  {
6      int Dem[10];
7      // Khởi tạo mảng Dem[i] = 0 mọi chỉ số i
8      ...
9      for ( i=0; i<n; i++)
10     {
11         x = a[i];
12         Dem[x]++;
13     }
14 }

```

Dĩ nhiên là các bạn có thể tối ưu thêm đoạn mã nguồn (code) trên.

2. THUẬT TOÁN “XOAY GƯƠNG”

Bài toán: Cho $a[n]$ là một mảng với phần gồm n phần tử. Ta cần chuyển m phần tử đầu tiên của mảng về cuối mảng mà không dùng mảng phụ.

Ví dụ:

Với $n = 8$ $a[8] = (1, 2, 3, 4, 5, 6, 7, 8)$

Nếu $m = 3$, thì kết quả là : $(4, 5, 6, 7, 8, 1, 2, 3)$

Nếu $m = 5$, thì kết quả là : $(6, 7, 8, 1, 2, 3, 4, 5)$

Nếu $m = 4$, thì kết quả là : $(5, 6, 7, 8, 1, 2, 3, 4)$.

Giải pháp:

Phương pháp thông thường các bạn thực hiện là dùng biến tạm để lưu phần tử đầu tiên $a[0]$, sau đó dồn các phần tử từ sau lên trước 1 phần tử. Thực hiện như vậy m lần.

```

5
4  main()
5  {
6      for ( i=0; i<m; i++)
7      {
8          tmp = a[0];
9          for ( j=0; j<n-1; j++)
10             a[j] = a[j+1];
11             a[n-1] = tmp;
12     }
13 }
14

```

Cách làm này hơi tốn kém... vì chúng ta cần $m*n$ phép tính (*gán....*)

Phương án đề xuất:

Các bạn có thể sử dụng thuật toán như sau:

B1: Xoay ngược mảng

B2: Xoay ngược m phần tử cuối

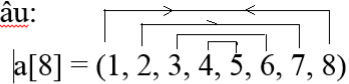
B3: Xoay ngược $(n-m)$ phần tử đầu

Cách làm này chỉ tốn cỡ n phép tính (*gán ..*)

Ví dụ:

Với $n = 8, m = 3$

Ban đầu:


 $a[8] = (1, 2, 3, 4, 5, 6, 7, 8)$

B1: $a[8] = (8, 7, 6, 5, 4, 3, 2, 1)$

B2: $a[8] = (8, 7, 6, 5, 4, 1, 2, 3)$

B3: $a[8] = (4, 5, 6, 7, 8, 1, 2, 3)$

Các bạn thử viết code cho ý tưởng trên và chứng minh thuật toán là đúng đắn.

3. VỀ NHÀ:

- Tìm hiểu thuật toán so khớp chuỗi Knuth–Morris–Pratt (hay thuật toán KMP)
- Buổi tiếp theo <http://forum.uit.edu.vn/threads/54127>
 - o STL (Standard Template Library (STL))
 - o Qui hoạch động
 - o Design Patterns