**TÌM KIẾM NHỊ PHÂN**

**BINARY SEARCHING**

**1.Phương pháp tìm kiếm:**

Thuật toán tìm kiếm nhị phân liên quan đến bài toán sau:

“ Cho mảng n phần tử đã được sắp tăng dần và một phần tử x. Tìm xem x có trong mảng hay không”

**Yêu cầu:** Thuật toán này chỉ có thể được dùng khi dãy số được sắp xếp đơn điệu theo thứ tự tăng hoặc giảm dần.

**Tư tưởng của thuật toán:** chọn phần tử ở vị trí giữa làm chốt, chia dãy thành 2 phần có kích thước nhỏ hơn. Sau đó so sánh phần tử cần tìm x với chốt, nếu x lớn hơn chốt tìm ở nửa sau của dãy, nếu x nhỏ hơn chốt tìm ở nửa trước của dãy (áp dụng với dãy tăng), quá trình trên tiếp tục cho tới khi tìm được x hoặc dãy chia không còn phần tử nào.

**Ví dụ:**

Cho dãy số: A={-6,1,3,5,8,10,14,16,19,21 }; x=5; dãy gồm 10 phần tử

Gọi phần tử chốt là k, ban đầu k=8

Bước 1: k=8, so sánh x với k, x<k ta tìm kiếm x ở nửa trước {6, 1,3,5,8}

Bước 2: k=3, so sánh x với k, x>k ta tìm kiếm x ở nửa sau {3,5,8}

Bước 3: k=5, so sánh x với k, x=k ta tìm được x kết thúc.

*Procedure TKNP (x: Item, a1,a2,...,an: Item);*

*Begin*

*d := 1; {d là điểm đầu của đoạn tìm kiếm}*

*c := n; {c là điểm cuối của đoạn tìm kiếm}*

*tim\_thay:=false;*

*while (d <=c) and not tim\_thay do*

*begin*

*g:= (d+c) div 2*

*if x = a[g] then tim\_thay :=true*

*else if x<a[g] then c := g-1*

*else d:=g+1*

*end*

*If tim\_thay then kq:=g*

*else kq := 0 {kq là vị trí của số hạng bằng x hoặc 0 nếu không tìm thấy x}*

*End;*

**2.Độ phức tạp :**

Để tìm kiếm một phần tử trong mảng. Với cách thông thường, ta phải duyệt tất cả các số từ a[1] đến a[n], tức là mất độ phức tạp O(n).

Tuy nhiên với mảng đơn điệu, ta dùng chặt nhị phân để tìm kiếm thì :

Ttốt= O(1) ( x nằm ở vị trí giữa mảng)

Txấu= O(logn)

Logarit là một hàm tăng chậm. Trong trường hợp ta còn băn khoăn về tính hiệu quả khi tìm kiếm nhị phân, hãy xét việc tìm kiếm một tên trong một cuốn danh bạ điện thoại có chứa một triệu tên. Tìm kiếm nhị phân cho phép ta tìm thấy bất kỳ tên nào chỉ sau nhiều nhất 21 lượt so sánh. Nếu ta có thể quản lý một danh sách có chứa tất cả mọi người trên thế giới được sắp xếp theo tên, ta có thể tìm thấy bất kỳ người nào trong vòng chưa đầy 35 bước.

**3.Bài tập vận dụng**

**A.Phần bài tập cơ bản**

**Bài 1. ĐOÁN SỐ**

Hai người chơi như sau: Người thứ nhất sẽ nghĩ ra một số nguyên dương trong khoảng từ 1 đến N (N được cho biết trước). Người thứ hai sẽ lần lượt đưa ra các số dự đoán. Với mỗi số dự đoán này, người thứ hai sẽ nhận được câu trả lời cho biết số mình vừa nêu ra lớn hơn, nhỏ hơn, hay bằng với số mà người thứ nhất đã nghĩ. Em hãy giúp người thứ hai chọn đúng số cần tìm với số lần đoán càng ít càng tốt.

**Thuật toán:**

Người thứ hai muốn chọn đúng số mà người thứ nhất nghĩ với số lần đoán ít nhất thì người thứ hai chắc chắn phải sử dụng đến thuật toán tìm kiếm nhị phân. Các bước sẽ lần lượt như sau:

*Bước 1.X:=1; y:=n; a:=0;*

*Bước 2.A:=(x+y) div 2*

*Bước 3.Lần đoán thứ i: a*

*Bước 4.Nếu a lớn hơn số cần tìm thì gán y:=a*

*Nếu a nhỏ hơn số cần tìm thì gán x:=a*

*Bước 5.Nếu số lần đoán vượt quá log2N thì chấm dứt*

*Ngược lại thì trở lại bước 2*

**Bài 2. BÀI TOÁN CỔ**

*"Vừa gà vừa chó*

*Bó lại cho tròn*

*Ba mươi sáu con*

*Một trăm chân chẵn*

*Hỏi có bao nhiêu gà bao nhiêu chó?"*

Bài toán này các em đều đã rất quen thuộc từ hồi học cấp I. Phương pháp để giải bài toán này là phương pháp ***giả thiết tạm***.

Tất cả có 36 con vật. Chúng không thể đều là gà vì như vậy sẽ chỉ có 72 chân. Cũng không thể nào là chó cả vì như vậy sẽ có cả thảy 144 chân (Số chân của chúng là 100 chân).

Áp dụng tư tưởng chặt nhị phân cho bài toán này như sau:

- Sắp xếp số gà theo thứ tự tăng dần (theo chiều từ dưới lên)

- Chia đôi tổng số gà, nếu tại điểm giữa đó tổng số chân gà và chân gà và chân chó lớn hơn 100 thì lấy nửa trên, và ngược lại lấy nửa dưới.

***Thuật toán:***

1. d:=0; c:=36; x:=100;

2.Trong khi d<c thì

2.1.g:= (d+c) div 2;

2.2.Nếu x=2\*g+4\*(36-g) thì

y:=c-g; Số gà là g; Số chó là y

2.3.Nếu x>2\*g+4\*(36-g) thì c:=g

2.4Nếu x<2\*g+4\*(36-g) thì d:=g

3.Quay lại bước 2

4.Kết thúc.

**Bài 3. SỐ 0 CUỐI CÙNG**

Cho một dãy số khoảng 1000000 kí tự số toàn 0 và 1. Biết rằng các số 0 đứng trước các chữ số 1: 000....0011...11. Hãy cho biết vị trí của số 0 cuối cùng trong dãy.

***Thuật toán:***

Ta tiến hành tìm kiếm nhị phân trên xâu kí tự để tìm ra vị trí số 0 cuối cùng như sau:

- Tìm phần tử giữa xâu đang xét

* So sánh kí tự ở vị trí giữa xâu với kí tự 0.
* Nếu kí tự giữa xâu là kí tự 0 thì ta tìm ở nửa sau của xâu, nếu không phải kí tự 0 (mà là 1) thì ta tìm ở nửa trước của xâu.

*1.d:=1; c:=length(s);*

*2. trong khi d<c thì*

*2.1 g:=(d+c+1) div 2;*

*2.2 Nếu s[g]='0' thì d:=g*

*2.3 Nếu s[g]='1' thì c:=g-1;*

*3.Quay lại bước 3*

*4.Kết thúc*

Vậy vị trí của số 0 cuối cùng trong dãy là *d*