

Thuật toán tìm kiếm nhị phân (Binary search)

Thuật toán Tìm kiếm nhị phân (Binary Search) là một thuật toán cao cấp hơn tìm kiếm tuần tự.

Đối với các dãy số lớn, thuật toán này tốt hơn hẳn so với các thuật toán tìm kiếm khác. Nhưng nó đòi hỏi dãy số phải được sắp xếp từ trước.

Tuy thuật toán binary search có ưu điểm là tìm kiếm nhanh, nhưng cũng có nhược điểm là nếu danh sách bị thay đổi, danh sách đó phải được sắp xếp lại trước khi tiến hành tìm kiếm tiếp. Và thao tác này thường tốn thời gian.

Thuật toán tìm kiếm nhị phân (Binary Search)

0123456789

Tìm số 23 trong Mảng

25812162338567291

L = 0

123456789

M = 4

25812162338567291

23 > 16

Lấy nửa sau với M = 7

L = 5 và R = 9

0123456789

L = 5

M = 7

R = 9

25812162338567291

23 < 56

Lấy nửa sau với M = 23

L = 5 và R = 6

0123456789

L = 5; M = 5

R = 6

25812162338567291

Tìm thấy giá trị 23

Vị trí số 5

Keep it Simple

DNM

Minh họa cho thuật toán tìm kiếm nhị phân, chúng ta cùng làm bài toán sau: Tìm vị trí phần tử có giá trị là **23** trong một mảng $A[2,5,8,12,16,23,38,56,72,91]$ đã được sắp xếp.

```
#include <iostream>

using namespace std;

// A recursive binary search function. It returns
// location of x in given array arr[l..r] is present,
// otherwise -1
int binarySearch(int arr[], int l, int r, int x)
{
    if (r >= l) {
```

```

    int mid = l + (r - l) / 2;

    // If the element is present at the middle
    // itself
    if (arr[mid] == x)
        return mid;

    // If element is smaller than mid, then
    // it can only be present in left subarray
    if (arr[mid] > x)
        return binarySearch(arr, l, mid - 1, x);

    // Else the element can only be present
    // in right subarray
    return binarySearch(arr, mid + 1, r, x);
}

// We reach here when element is not
// present in array
return -1;
}

int main(void)
{
    int arr[] = { 2, 3, 4, 10, 40 };
    int x = 10;
    int n = sizeof(arr) / sizeof(arr[0]);
    int result = binarySearch(arr, 0, n - 1, x);
    (result == -1) ? cout << "Element is not present in array"
                  : cout << "Element is present at index " << result;

    return 0;
}

```

Revision #1

Created 6 May 2023 04:03:26 by Laptrinh.vn

Updated 27 June 2023 09:20:43 by Laptrinh.vn