

Sequential Vs Binary Search

โดย ผศ. ดร. จรัสศรี รุ่งรัตนอุบล

สอวน. ค่าย 2

13 มีนาคม 2562

Search

search หรือการค้นหาเป็น **อัลกอริทึมหรือขั้นตอนวิธี** ที่จะมีการเรียกใช้บ่อย เพราะเรามักจะต้องสืบค้นข้อมูลเป็นประจำ โดยเฉพาะข้อมูลในอาร์เรย์

การสืบค้นจะมีรูปแบบฟังก์ชันดังนี้

```
int search(array, element)
```

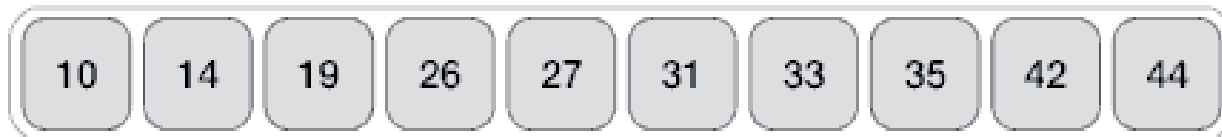
โดยอาจจะส่งคำตอบ

- เป็น **true** หรือ **false**
- เป็น **ตำแหน่งหรืออินเด็กซ์** ของค่านั้นในอาร์เรย์ หรือส่ง -1 ถ้าไม่เจอค่านี้

Sequential or Linear Search

เป็นการค้นหาแบบลำดับโดยจะเปรียบเทียบค่าไปที่ละตำแหน่ง

Linear Search



=
33

seqSearch()

- จะรับ อาร์เรย์ และ ค่าที่ต้องการหา
- ถ้าพบค่าจะส่งตำแหน่งของค่านั้นในอาร์เรย์ ไม่เช่นนั้นจะส่ง -1 ออกมา

Binary search

- จะรับ อาร์เรย์ที่เรียงแล้ว และ ค่าที่ต้องการหา
- ถ้าพบค่าจะส่งตำแหน่งของค่านั้นในอาร์เรย์ ไม่เช่นนั้นจะส่ง -1 ออกมา

Search for 47

| | | | | | | | | |
|---|---|---|----|----|----|----|----|----|
| 0 | 4 | 7 | 10 | 14 | 23 | 45 | 47 | 53 |
|---|---|---|----|----|----|----|----|----|

binSearch()

```
function binary_search(A, n, T):  
    L := 0  
    R := n - 1  
    while L <= R:  
        m := floor((L + R) / 2)  
        if A[m] < T:  
            L := m + 1  
        else if A[m] > T:  
            R := m - 1  
        else:  
            return m  
    return unsuccessful
```

Binary search

steps: 0



| | | | | | | | | | | | | | | | | |
|-----|---|---|---|----|----|----|----|-----|----|----|----|----|----|----|----|------|
| 1 | 3 | 5 | 7 | 11 | 13 | 17 | 19 | 23 | 29 | 31 | 37 | 41 | 43 | 47 | 53 | 59 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| Low | | | | | | | | mid | | | | | | | | high |

Sequential search

steps: 0



| | | | | | | | | | | | | | | | | |
|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 3 | 5 | 7 | 11 | 13 | 17 | 19 | 23 | 29 | 31 | 37 | 41 | 43 | 47 | 53 | 59 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |

www.mathwarehouse.com

Finding the leftmost / rightmost

- ถ้าเรามีอาร์เรย์ดังนี้ {1,2,3,4,4,4,5,6,7}
- คำตอบที่ได้จาก `binSearch(arr, 4)` คือ
- ในกรณีที่ข้อมูลมีค่าซ้ำกันในบางครั้งเราต้องการให้ตอบค่า
ทางซ้ายมือสุด หรือค่าทางขวามือสุด

Finding the leftmost Algo.

{1,2,3,4,4,4,5,6,7}

```
function binary_search_leftmost(A, n, T):  
    L := 0  
    R := n  
    while L < R:  
        m := floor((L + R) / 2)  
        if A[m] < T:  
            L := m + 1  
        else:  
            R := m  
    return L
```

Finding the rightmost Algo.

{1,2,3,4,4,4,5,6,7}

```
function binary_search_rightmost(A, n, T):  
    L := 0  
    R := n  
    while L < R:  
        m := floor((L + R) / 2)  
        if A[m] <= T:  
            L := m + 1  
        else:  
            R := m  
    return L - 1
```

Problem 1: FindTheLast

- 5 คือขนาดของอาร์เรย์ 1 คือค่าที่ต้องการหา
- ข้อมูลทั้ง 5 ค่า

| SAMPLE INPUT | SAMPLE OUTPUT |
|------------------|---------------|
| 5 1 1 2 3 4 1 | 5 |

- คำตอบคือ ค่าที่ต้องการหาที่อยู่ตำแหน่งหลังสุด

Problem 2: Min-Max

- ให้เขียนโปรแกรมเพื่อหาผลรวมน้อยที่สุดและมากที่สุด Min-Max

| SAMPLE INPUT | SAMPLE OUTPUT |
|----------------|---------------|
| 5 1 2 3 4 5 | 10 14 |

- **Explanation**
- Our initial numbers are 1,2,3,4 and 5. We can calculate the following sums using four of the five integers:
- If we sum everything except 1, our sum is $2+3+4+5=14$.
- If we sum everything except 2, our sum is $1+3+4+5=13$.
- If we sum everything except 3, our sum is $1+2+4+5=12$.
- If we sum everything except 4, our sum is $1+3+4+5=11$.
- If we sum everything except 5, our sum is $1+2+3+4=10$.
- As you can see, the minimal sum is $1+2+3+4=10$ and the maximal sum is $2+3+4+5=14$.

Problems

- <https://www.hackerearth.com/practice/algorithms/searching/linear-search/practice-problems/>