# Search

Canberra Institute of Technology

aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

# Contents

- The search problem?
- Linear search
- Binary search

# The Search Problem

- Given a set to search, find the element or elements that match a given set of properties.
- The set to search can be explicit, such as an array of enemies currently alive
- Or implicit, such as all legal chess moves from a given starting position.

# Key-Value pairs

- The data you have before searching is called a key and the data you get back is called the value.

- In the simplest case, you simply search to find if the key exists in the set you are searching.

# Searching through explicit data

- The simplest form of search is searching for an element with a specific value in an array of elements.

- This could include:
  - Finding the KD ratio of a player with a specific name in a multiplayer game.
  - Finding what kind of tile is at a specific XY location in the world
  - Finding the enemy with the lowest health
  - Finding if an image file with a given filepath is already in the loaded textures list
  - Checking to see if the player has a specific item in their inventory.

# Linear Search

- Linear search is the simplest solution.
- You just start at the first element and check every possibility one at a time, checking if each matches your criteria.
- In an array this is a for loop that iterates over every element in the array.

# Linear Search

```
FUNCTION LinearSearch(arr, len, key)
    FOR i = 0 .. len
        IF arr[i] == key
            return arr[i]
        END IF
    END FOR
    RETURN NULL
END FUNCTION
```

# Binary Search

- Linear search, as its name suggests, runs in linear O(n) time.
  - If you triple how many elements are in the list, the algorithm takes 3 times as long (in the worst case).

- While it might be unintuitive, we can do significantly better than linear

# Binary Search

- Question:
  - I am thinking of a number between 1 and 100, inclusive. When you make a guess I will tell you if the number is larger or smaller than your guess.
  - What is the most efficient way to guess the number?

# Binary Search

- This is how binary search works
  - When you start searching, the element you're looking for could be any element, so the minimum possible element is the first one and the maximum is the last.

  - Check the element half way between the minimum and the maximum.
    - If it is the element you're looking for, you've found it!
    - If it is smaller than the element you're looking for, then the element MUST be after it, so the new minimum is one past the element you're checking.
    - If it is bigger than the element you're looking for, then the element MUST be before it, so the new maximum is one before the element you're checking.

  - Repeat

- Binary search requires a list that is already in order from smallest to biggest.

# Binary Search

- Search for the number 41

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 7 | 8 | 12 | 16 | 31 | 32 | 35 | 41 | 49 | 50 | 51 | 61 |

# Binary Search

Min possible element

Middle element

Max possible element

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 7 | 8 | 12 | 16 | 31 | 32 | 35 | 41 | 49 | 50 | 51 | 61 |

Canberra Institute of Technology

aie

SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

# Binary Search

Min possible element

Middle element

Max possible element

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 7 | 8 | 12 | 16 | 31 | 32 | 35 | 41 | 49 | 50 | 51 | 61 |

41 > 32

# Binary Search

Min possible element

Max possible element

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 7 | 8 | 12 | 16 | 31 | 32 | 35 | 41 | 49 | 50 | 51 | 61 |

41 > 32

# Binary Search

Min possible element

Max possible element

Middle element

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 7 | 8 | 12 | 16 | 31 | 32 | 35 | 41 | 49 | 50 | 51 | 61 |

Canberra Institute of Technology

aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

# Binary Search

Min possible element

Max possible element

Middle element

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 7 | 8 | 12 | 16 | 31 | 32 | 35 | 41 | 49 | 50 | 51 | 61 |

41 < 50

# Binary Search

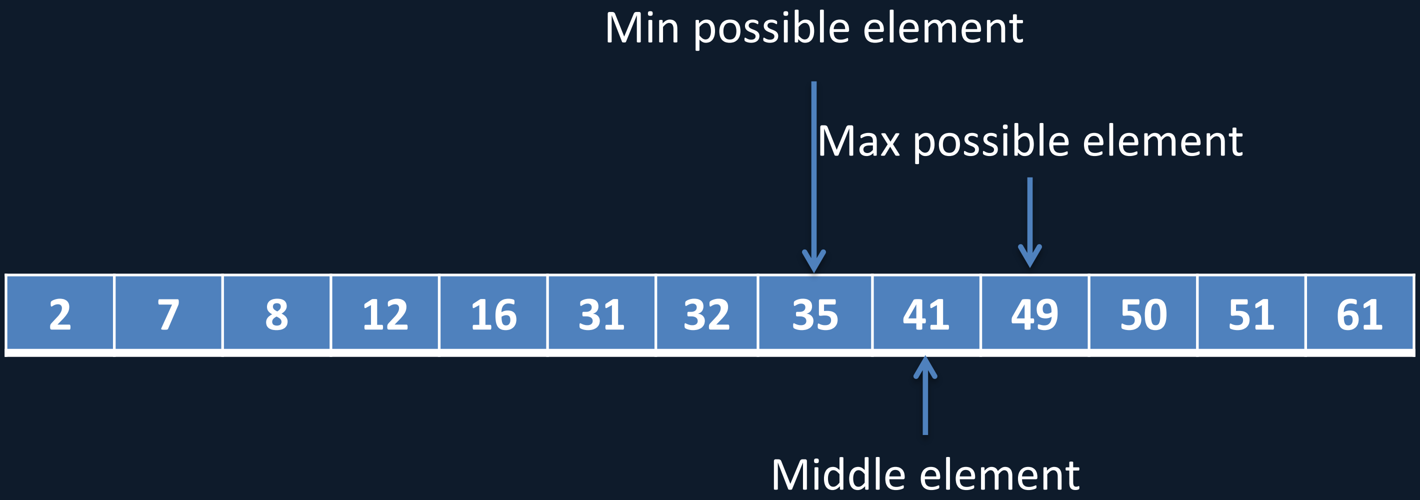
Min possible element

Max possible element

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 7 | 8 | 12 | 16 | 31 | 32 | 35 | 41 | 49 | 50 | 51 | 61 |

41 < 50

# Binary Search



# Binary Search

Min possible element

Max possible element

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 7 | 8 | 12 | 16 | 31 | 32 | 35 | 41 | 49 | 50 | 51 | 61 |

Middle element

41 == 50

# Binary Search

```
FUNCTION BinarySearch(arr, len, key)
    max = len - 1
    min = 0

    WHILE max != min
        middle = (min + max) / 2

        IF arr[middle] == key
            RETURN arr[middle]
        ELSE IF arr[middle] > key
            max = middle – 1
        ELSE IF arr[middle] < key
            min = middle + 1
        END IF
    END WHILE
    RETURN NULL
END FUNCTION
```

Canberra Institute of Technology

aie
SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

# Binary Search

- Binary Search is far more efficient than linear search in almost all cases.

- Binary Search runs in O( log n ) time
  - This means if you double the size of the array, you only need to make one more check to find a given key.
  - If your array has 60,000 elements, in the worst case, you need 16 checks.
    - Double the array to 120,000 elements, the worst case becomes 17 checks

# Summary

- We’ve covered the very basics of searching for data
  - Linear and Binary search
- There many, many more ways to organize and then search through your data. We will cover much more in later lectures.
  - Binary Trees, red-black trees, hash tables, heaps

# References


- Sedgewick, R, and Wayne, K "*Algorithms*", 4th Ed, Chp 3, Addison-Wesley (2011)