



Searching Algorithms

GCSE Student Booster

Key Information

- 1) Remember this booster is here to **help you**. Please consider your behaviour in the chat.
- 2) If you are in a room with a teacher/group, please login to the meeting. This is so we can mark your attendance. This information goes into a **prize draw**.
- 3) Make sure the name on the meeting is the **SAME** as the name on your Isaac account. We can't mark you present if they don't match.



Intended Learning Outcomes

By the end of this session, you will be able to:

- Use the key steps to perform a Linear search
- Use the key steps to perform a Binary search
- Explain when a Linear or Binary search can or cannot be used
- Explain how effective the methods are for different sized data sets



Types of searches

Linear
searches

Binary
searches

Steps of a Linear Search

Performing a linear search involves the following steps:

1. Take a list of data and a search item.
2. Starting from the first position, repeat steps 3–5 until finding the search item or reaching the end of the list:
3. Compare the item at the current position to the search item.
4. If the current item is equal to the search item, stop searching.
5. Otherwise, move to the next position in the list.

Example

Linear Search

Step 1: Take a list of data and a search item.

1	10	9	2	4	5	6	8
---	----	---	---	---	---	---	---

Card we are looking for

4

Linear Search

Example

Add the index, starting at 0 (index = position within the data)

0	1	2	3	4	5	6	7
1	10	9	2	4	5	6	8

Card we are looking for

4



Linear Search

Example

starting at 0 (index = position within the data)

0	1	2	3	4	5	6	7
1	10	9	2	4	5	6	8

4

Steps 2 to 5 IF card == cardList[0] THEN
Print (Found)
EL SE
move to the next card (or index = index +1)



Linear Search

Example

0	1	2	3	4	5	6	7
1	10	9	2	4	5	6	8

4

Steps 2 to 5 IF card == cardList[1] THEN
Print (Found)
EL SE
move to the next card (or index = index +1)



Linear Search

Example

0	1	2	3	4	5	6	7
1	10	9	2	4	5	6	8

Steps 2 to 5 IF card == cardList[2] THEN
 Print (Found)
 ELSE
 move to the next card (or index = index +1)

Linear Search

Example

0	1	2	3	4	5	6	7
1	10	9	2	4	5	6	8

Steps 2 to 5 IF card == cardList[3] THEN
 Print (Found)
 ELSE
 move to the next card (or index = index +1)

Linear Search

Example

0	1	2	3	4	5	6	7
1	10	9	2	4	5	6	8

Steps 2 to 5 IF card == cardList[4] THEN
 Print (Found)
 ELSE
 move to the next card (or index = index +1)

FOUND in cardList[4]
index

Pseudocode

```

1  index = 0
2  match = False

3  WHILE match == False AND index < cardList.length - 1
4      IF card == cardList[index] THEN
5          PRINT ("Match")
6          match = True
7      ELSE
8          Index = index + 1
9      END IF
10 END WHILE

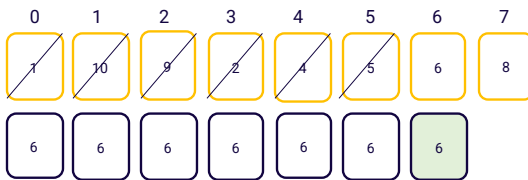
```



Linear Search

Let's do this together

How many searches to find the number 6?



Steps 2 to 5 IF card == cardList[index] THEN
 Print (Found)
 ELSE
 move to the next card (or index = index +1)



Linear Search

Works the same with words

Step 1: Take a list of data and an item that is being searched for (the search item)



Card we are looking for

Green



Linear Search

Works the same with words

~~Pink~~

Red

Blue

Green

Grey


Black

Yellow

Orange

Green

```
IF card == cardList[index] THEN
  Print (Found)
ELSE
  move to the next card (or index = index +1)
```



Linear Search

Works the same with words

~~Pink~~

~~Red~~

Blue

Green

Grey


Black

Yellow

Orange

Green

```
IF card == cardList[index] THEN
  Print (Found)
ELSE
  move to the next card (or index = index +1)
```



Linear Search

Works the same with words

~~Pink~~

~~Red~~

~~Blue~~

Green

Grey


Black

Yellow

Orange

Green

```
IF card == cardList[index] THEN
  Print (Found)
ELSE
  move to the next card (or index = index +1)
```



Linear Search

Works the same with words



```
IF card == cardList[index] THEN
  Print (Found)
ELSE
  move to the next card (or index = index +1)
```



Student Led Task

Complete activities in "Handout 1 Task 1"

Searching Algorithms

Handout 1 – Searching Algorithms: Student led tasks

Task 1 – Linear search

1. Describe the steps that a linear search would take to find Simon in the below data set.

Fred	Bob	Sally	Simon	John	Amy
------	-----	-------	-------	------	-----



Linear search characteristics

- The only methodical way to find a specific item in an **unordered** list of items
- Look at every item in the list, one after another, and check if it is what you are looking for (starting at index 0).
- Once item is found the search ends
- Simplest search method when compared to Binary search
- Less efficient on larger data sets compared to Binary search



Binary Search

- Binary search is a highly efficient algorithm for searching, but it requires an **ordered** list of items.
- The list can be ordered in **ascending** (low to high) or **descending** (high to low) order, and the algorithm needs to be adjusted accordingly to focus on the correct range of items.



Steps of a Binary Search

Performing a binary search involves these steps:

1. Use an ordered list of data and a search item.
2. Repeat steps 3–7 until finding the search item or no items left to check:
 3. Find the item at the midpoint position.
 4. Compare the item at the midpoint to the search item.
 5. If equal, stop searching.
 6. If less, focus on items after the midpoint for next search
 7. If greater, focus on items before the midpoint for next search



Pseudocode

```

1. found = False
2. found_index = -1
3. left = 0
4. right = LEN(items) - 1

5. WHILE left <= right AND found == False
6.     midpoint = (left + right) DIV 2
7.     IF items[midpoint] == search_item THEN
8.         found_index = midpoint
9.         found = True
10.    ELSEIF items[midpoint] < search_item THEN
11.        left = midpoint + 1
12.    ELSE
13.        right = midpoint - 1
14.    ENDIF
15. ENDWHILE
  
```

You will need to recognise the code, but not remember it for your exam



Binary Search

Example

Add the index, starting at 0 (index = position within the data)

0	1	2	3	4	5	6	7
1	2	3	4	5	6	7	8

Card we are looking for

7



Binary Search

Example

Add the index, starting at 0 (index = position within the data)

0	1	2	3	4	5	6	7
1	2	3	4	5	6	7	8

$$0 + 7 = 7$$

$$7 // 2 = 3$$

Mid point = 3 (Integer)

7



Binary Search

Example

Add the index, starting at 0 (index = position within the data)

0	1	2	3	4	5	6	7
1	2	3	4	5	6	7	8

$$4 + 7 = 11$$

$$11 // 2 = 5$$

Mid point = 5 (Integer)

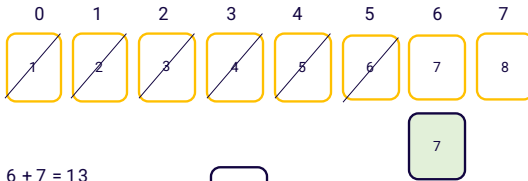
7



Binary Search

Example

Add the index, starting at 0 (index = position within the data)



$$6 + 7 = 13$$

$$13 // 2 = 6$$

Mid point = 6 (Integer)



Binary Search Characteristics

- Only works on **sorted** data sets
- Finds the middle position (Divide and Conquer method)
- Checks the middle value against the search criteria
- Casts off half the values in the list, recalculates the middle position and runs again until value is found.
- If it list it finds it
- Efficient on large data sets than Linear Search



Activity

Complete activities in "Handout 1 Task 2"

Task 2 - Binary search

1. Show the stages of a binary search to find the word **Terry** when applied to the data shown.

Alice	Fred	Geraint	Loise	Ravina	Sam	Steven	Terry	Wilma
-------	------	---------	-------	--------	-----	--------	-------	-------

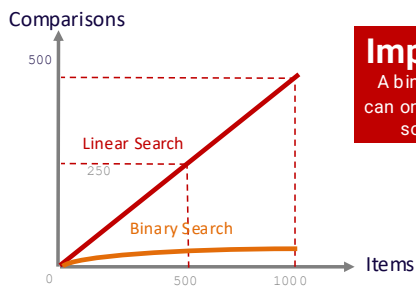


Efficiency

	Binary search	Linear Search
Items in list	Maximum comparisons	
10	4	10
100	7	100
1000	10	1000
10,000	14	10,000
100,000	17	100,000
1,000,000	20	1,000,000



Binary Search - Doing Stuff Really Well



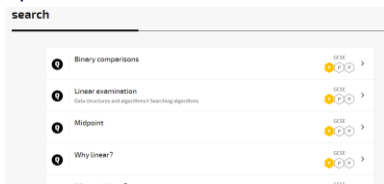
Important

A binary search can only work on a sorted list




Isaac Computer Science

Complete "Handout 1 Task 3 – Gameboard"



<https://isaacscs.org/assignment/9c661cf8-3cbb-41b8-a7a9-7e604041c77c>



 Padlet link

Student Led Task


What are the characteristics of each searching algorithm?

Linear Search	Binary Search

Student Led Task

What are the characteristics of each searching algorithm?

Linear Search	Binary Search
<ul style="list-style-type: none"> • Simplest search method • Scans one item at a time against the search criteria • Once item is found the search ends • The data set <u>does not</u> need to be ordered. • Less efficient on larger data sets 	<ul style="list-style-type: none"> • Only works of ordered data sets • Finds the middle position (Divide and Conquer method) • Checks the middle value against the search criteria • Casts off half the values in the list, recalculates the middle position and runs again until value is found. • Efficient on large data sets than Linear Search

 Padlet link

Student Led Task

What are the advantages and disadvantages of a linear and a binary search?

Student Led Task

Linear search

Advantage	Disadvantage
Works on any data set, sorted or unsorted.	Less efficient than a binary search, may have to check all the items in the array to find the search criteria.



Student Led Task

Binary search

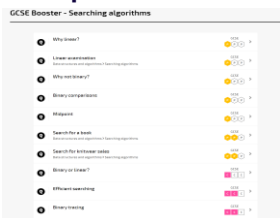
Advantage	Disadvantage
More efficient method of searching an ordered list than a linear search, much less items on average need to be checked than with the linear search.	List must be ordered Can be inefficient if the item you want is at the start of the list



Isaac Gameboard practice

- If you want more searching algorithms practice, then try this gameboard.

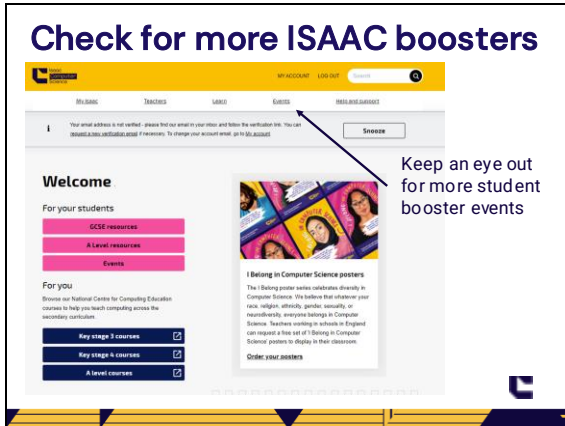
- You will need to sign in to **Isaac Computer Science** or register for a free account if not done already.



ncce.io/isc-search



Check for more ISAAC boosters



The screenshot shows the ISAAC website with a yellow header. The main content area has a 'Welcome' section with links for 'For your students' (GCSE resources, A Level resources, Events) and 'For you' (Browse our National Centre for Computing Education courses, Key stage 3 courses, Key stage 4 courses, A Level resources). A featured poster titled 'I Belong in Computer Science posters' is displayed, with a callout box saying 'Keep an eye out for more student booster events'.

Intended Learning Outcomes

By the end of this session, you will be able to:

- Use the key steps to perform a Linear search
- Use the key steps to perform a Binary search
- Explain when a Linear or Binary search can or cannot be used
- Explain how effective the methods are for different sized data sets

Isaac Computing

Link to Searching Algorithms topic on the Isaac Computer Science website

<https://isaacomputerscience.org/topics/searching?examBoard=all&stage=all>

"Education is the passport to the future, for tomorrow belongs to those who prepare for it today."

Malcolm X



Thank you