Binary or linear search? 2

Practice 1

Identify **three** reasons why you may choose to perform a linear search rather than a binary search on a list of data.

- [ ] The list is unsorted
- [ ] The list is sorted
- [ ] The list is very short
- [ ] The list is very long
- [ ] The algorithm is simpler to implement

Binary search: max comparisons 3

Practice 1

A local hockey club keeps a directory of players' email addresses, which they use to send out information about training and fixtures. It currently has 1,000 entries that are sorted in order.

What is the maximum number of comparisons a binary search would have perform to find a specific player's email address?

Quiz:
STEM SMART 2025 Residential Catch Up Quiz 2 - Programming

Searching algorithms

Challenge 1

LLM marked question

Computers frequently need to search through lists of data. This happens, for example, when you try to find a file with a particular name on your computer, or when you enter keywords into a search engine to find websites on the internet.

State one characteristic of a linear search and one characteristic of a binary search.

[2 marks]

Quiz:
STEM SMART 2025 Residential Catch Up Quiz 2 - Programming

Bubble sort: complete 1

Challenge 2

Kate has written some pseudocode to sort an array of data into **ascending** order. The algorithm she wrote is a version of bubble sort.

The **first three** statements and the **last two** statements of the algorithm are presented below:

Pseudocode

```
ARRAY items = [3, 8, 1, 10, 23, 78, 12]
num_items = LEN(items)
temp = 0

>>>>>  missing code statements  <<<<<<

    ENDFOR
ENDFOR
```

Drag and drop **the rest of the statements** into the correct order to complete the pseudocode, paying careful attention to indentation as you go.

Available items

```
ENDIF
```

```
items[index + 1] = temp
```

```
items[index] = items[index + 1]
```

```
FOR index = 0 TO num_items - 2
```

```
IF items[index] > items[index + 1] THEN
```

```
temp = items[index]
```

```
FOR pass_number = 1 TO num_items - 1
```

Quiz:
STEM SMART 2025 Residential Catch Up Quiz 2 - Programming

Bubble sort: maximum swaps

Practice 1

Michael is going to sort an array `items` that holds **5 random numbers** generated using a subroutine that produces random integers. As the numbers are generated at runtime, it is not possible to guess how the unsorted array will look.

The array is to be sorted into ascending order using the version of the bubble sort algorithm shown in pseudocode below:

Pseudocode

```
PROCEDURE bubble_sort(items)
    // Initialise variables
    num_items = LEN(items)
    temp = 0
    pass_number = 1
    swapped = True

    // Continue while swaps have been made and there are more passes to evaluat
    WHILE swapped == True AND pass_number <= num_items - 1
        swapped = False
        FOR index = 0 TO num_items - 2
            // Check if items are out of order
            IF (items[index] > items[index + 1]) THEN
                // Swap items
                temp = items[index]
                items[index] = items[index + 1]
                items[index + 1] = temp
                swapped = True
            ENDIF
            pass_number = pass_number + 1
        NEXT index
    ENDWHILE
ENDPROCEDURE
```

Part A

What is the **maximum** number of **passes** that might be performed?

Part B

What is the **maximum** number of **swaps** that might be carried out?

Quiz:
STEM SMART 2025 Residential Catch Up Quiz
2 - Programming

Insertion: trace 1

Practice 2

Pam is a teacher. She wants to sort her students test scores so that the highest score appears at the start of the list.

She uses an **insertion sort** to sort the data.

Drag and drop the lines into the correct order to show the state of the list of scores as the data is progressively sorted (i.e. as each item is correctly positioned).

The initial order of the list is: $43, 74, 64, 68, 49, 70$

Available items

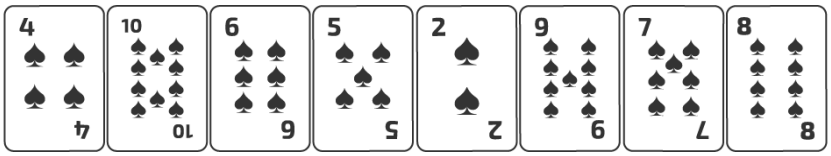
- 74, 68, 64, 43, 49, 70
- 74, 43, 64, 68, 49, 70
- 74, 70, 68, 64, 49, 43
- 74, 64, 43, 68, 49, 70
- 74, 68, 64, 49, 43, 70

Quiz:
STEM SMART 2025 Residential Catch Up Quiz 2 - Programming

Merge sort: trace 4

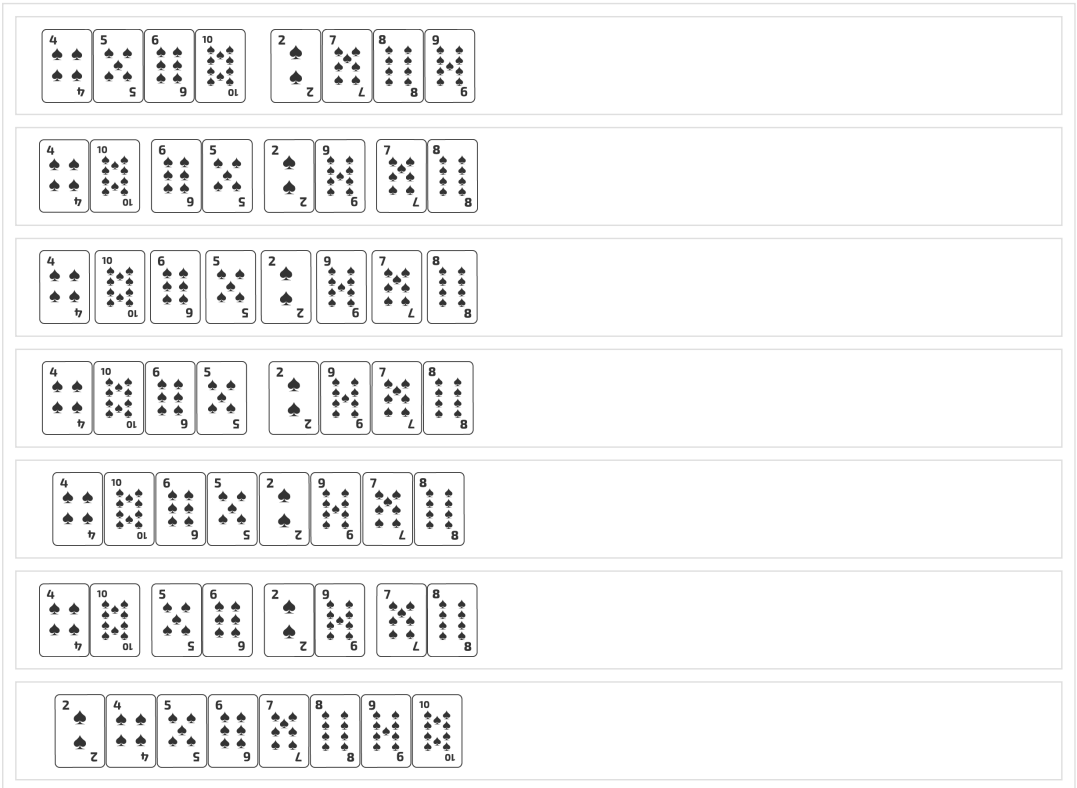
Practice 2

You have been given some playing cards that have not been sorted:



Drag the lines of cards below into the correct order to show how **each** stage of a merge sort would be performed to sort the the data from lowest to highest.

Available items



Quiz:
STEM SMART 2025 Residential Catch Up Quiz 2 - Programming

Dictionaries: purpose of the key

Practice 1

In a dictionary data structure, data is stored in key/value pairs.

What is the purpose of the key?

- To specify the order of the items
- To store metadata about the item
- To access a specific value associated with it
- To determine the data type of the value

Quiz:
STEM SMART 2025 Residential Catch Up Quiz 2 - Programming

Recursion: trace the code 1

Challenge 2

A recursive subroutine has been written as follows:

Pseudocode

```
FUNCTION do_something(x, y)
    IF x == 1 THEN
        RETURN y
    ELSE
        RETURN do_something(x-1, x+y)
    ENDIF
ENDFUNCTION
```

Trace the subroutine to determine what the final return value will be when the following call is made:

```
do_something(5, 2)
```

Quiz:
STEM SMART 2025 Residential Catch Up Quiz 2 - Programming

Features of polymorphism

Challenge 1

LLM marked question

In object-oriented programming, polymorphism is a key concept that allows objects to be treated as instances of their parent class.

State two features of polymorphism.

[2 marks]