LT12d Merge Sort

ALGORITHM

Unplugged Activity:

1. Start with a stack of 10 cards, all facing down on the table.
2. Divide the cards into 2 groups, then sub-divide the 2 groups again into further 2 groups … Until every group contain only 1 card.
3. Compare the values of cards between the 2 ‘neighbouring’ groups, and merge them back together. Repeat this until only 1 group left.
4. The cards in the group is sorted.

Description :

The **Merge Sort** algorithm will first divide the list into half over and over again until each list contains only one element.

Then it will merge two lists together into a sorted list repeatedly until all the elements are combined together in one single sorted list.

Example :

Unsorted array: [5, 2, 1, 8, 9]

- Step 1: Start by dividing the array into two, again and again until every array contain only one element.

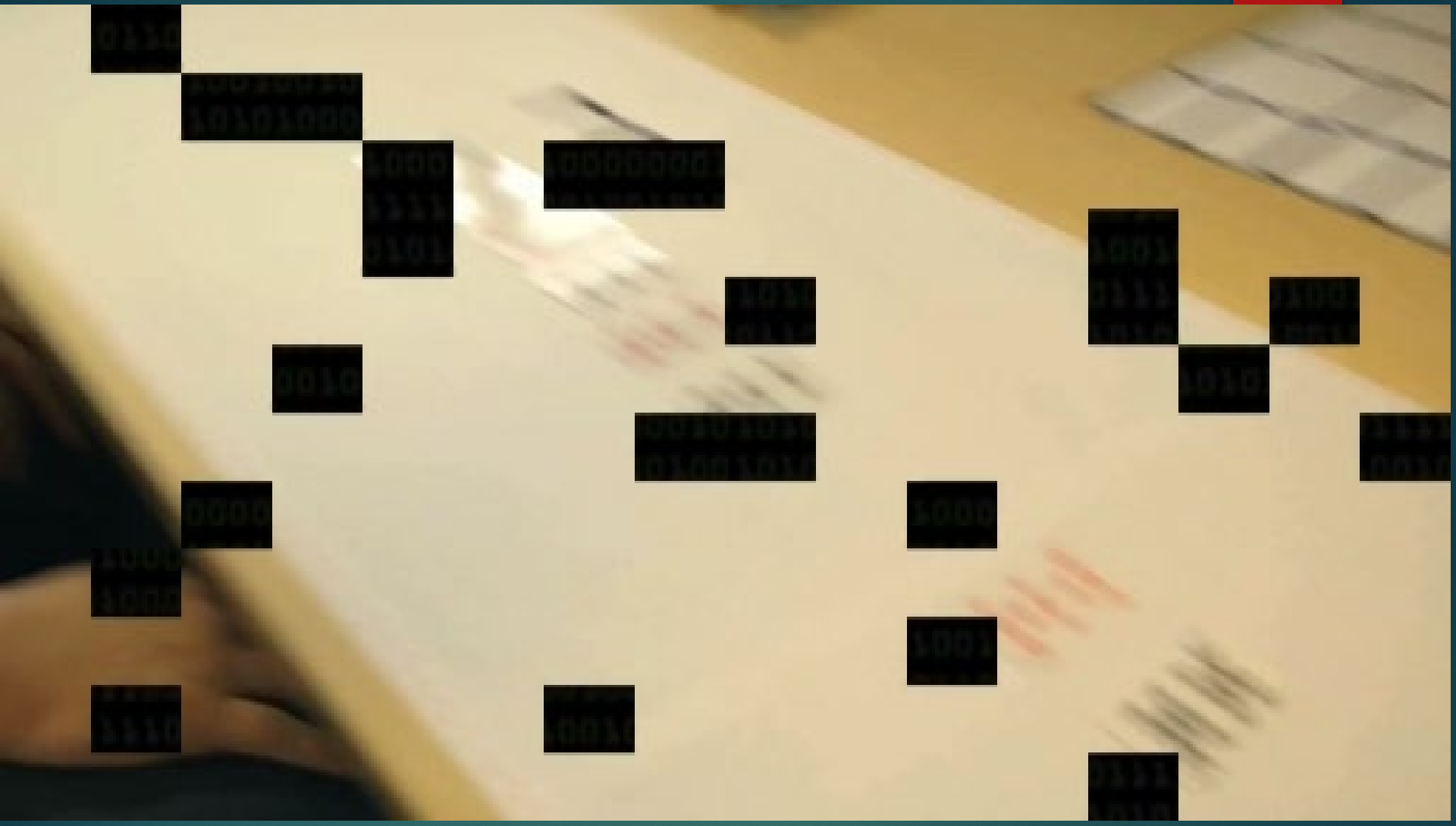
```
[5, 2, 1, 8, 9] -> [5, 2] [1, 8, 9]
-> [5] [2] [1] [8, 9]  -> [5] [2] [1] [8] [9]
```

- Step 2: Merge them back in sorted order:

```
-> [2, 5] [1] [8, 9]
-> [2, 5] [1, 8, 9]
-> [1, 2, 5, 8, 9]
```

Basic Skills :

1. Split a list recursively
2. Merge two lists together



Extracted from “Getting Sorted & Big O Notation – Computerphile”
https://www.youtube.com/watch?v=kgBjXUE_Nwc

Merge Sort : Order of Growth

Best Case : **O(n** log **n)**

Worst Case : **O(n** log **n)**

Average Case : **O(n** log **n)**