
Selection
Sort

A card for you to write:

Selection Sort

- Repeatedly finds the largest number and swaps it into place.
- O(n^2) in all cases. There is no best case.
- It works for all kinds of data, Strings, chars, doubles…

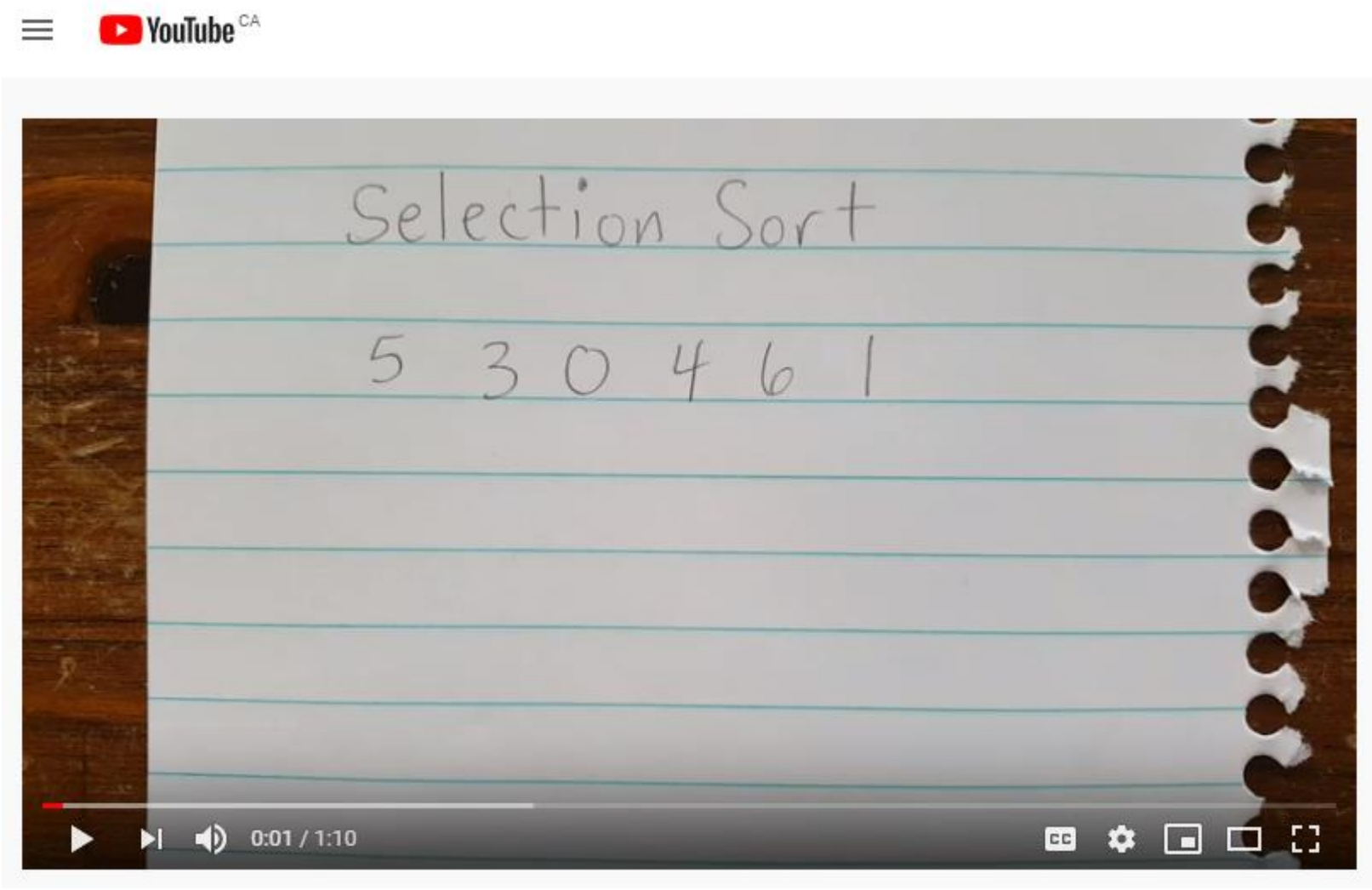
A card for you to write:

Selection Sort Tradeoff

(+) Selection Sort is simple to understand and is simple to code.

(-) It's simplicity means it isn't very clever or efficient. That means it is SLOW.

I posted a YouTube video Tracing Selection Sort:



https://www.youtube.com/watch?v=NiKCsHd2K70

A very important application of Selection Sort:

"Children are perhaps the greatest advocates of selection sort. Every October, Halloween candies are consumed from best to worst. Whether daily sampling is limited or not, it is clear that choices of the next treat consumed are based on 'the next biggest piece' or 'the next most favorite' and so on. Children consume treats in decreasing order of acceptability.

Similarly, when we select plants from a greenhouse, check produce in the store or pick strawberries from the farm, we seek the best items first."

Bailey, Duane. Java Structures. 1999. Pg 80.

To translate it to English…

```
int a[] = {6, 4, 3, 7, 8, 1, 0, 2};
for (int left = a.length - 1 ; left > 0 ; left--)
{
    int max = 0;
    for (int i = 1 ; i < left ; i++)
    {
        if (a [max] < a [i])
            max = i;
    }
    int temp = a [max];
    a [max] = a [left - 1];
    a [left - 1] = temp;
}
```

To translate it to English…

```
Declare the array
for (int left = a.length - 1 ; left > 0 ; left--)
{
    int max = 0;
    for (int i = 1 ; i < left ; i++)
    {
        if (a [max] < a [i])
            max = i;
    }
    int temp = a [max];
    a [max] = a [left - 1];
    a [left - 1] = temp;
}
```

To translate it to English…

```
Declare the array
for (int left = a.length - 1 ; left > 0 ; left--)
{
    Find the max
    int temp = a [max];
    a [max] = a [left - 1];
    a [left - 1] = temp;
}
```

To translate it to English…

```
Declare the array
for (int left = a.length - 1 ; left > 0 ; left--)
{
    Find the max
    Swap the max into place
}
```

To translate it to English…

```
Declare the array
For the non-sorted part of the array
{
    Find the max
    Swap the max into place
}
```


Tracing

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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