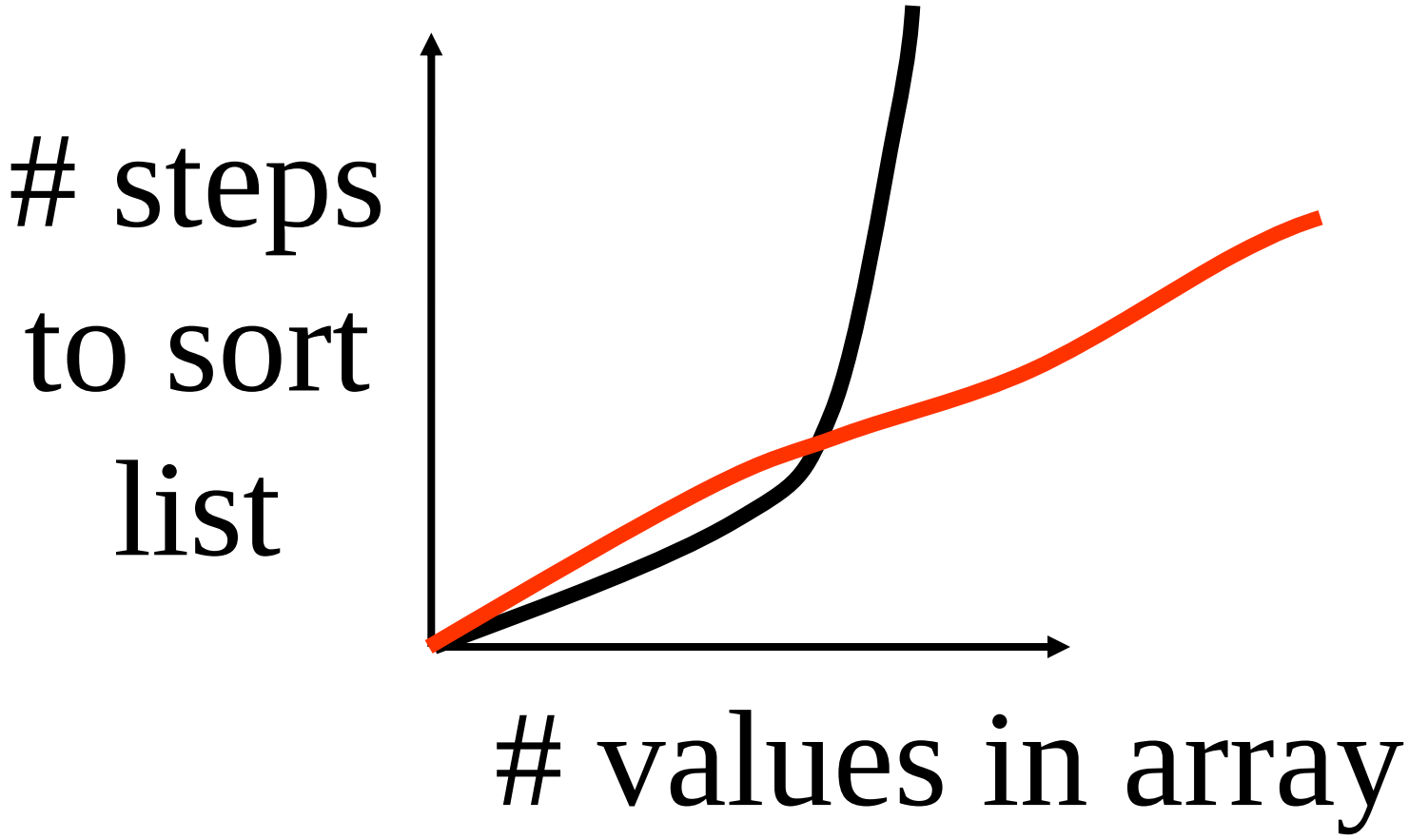
Step Into Java: Merge Sort

Mr. Neat

Java

Quadratic sorting algorithms are nice but...



Merge Sort Pseudo Code

```
void mergeSort(int A[], int first, int last)
{
        // find middle index of A
        // sort the first half of A
        // sort the second half of A
        // merge the first and second halves of A
}
```

Let's look at merge...

```
/*
precondition: lists A and B are sorted in
                    non-decreasing order
postcondition: list C contains all the
                        values from lists A and
                        B in nondecreasing
                        order
*/

void merge(int A[], int B[], int C[])
```

List A

3	11	17	19	24	29	31	37

List B

1	4	5	15	18	25	27	36

How many elements does List C have?

Pseudo Code for Merge

A) List A is done, get value from List B
B) List B is done, get value from List A
C) Neither is done, if List A[i] < B[k],
then get value from List A
D) Neither is done, if List B[k] <= List A[i]
then get value from List B

List A

3	11	17	19	24	29	31	37

List B

1	4	5	15	18	25	27	36

Let’s count which rules we use...

Merge Sort Pseudo Code

```
void mergeSort(int A[], int first, int last)
{
        // find middle index of A
        // sort the first half of A
        // sort the second half of A
        // merge the first and second halves of A
}
```