

# Yixuan's CS333 Project 4 (Extension) - Ruby

## Extension 1:

### Implements a generic quicksort program in Ruby.

Ruby is a dynamically typed language, so I don't need to specify the type of array when implementing the program.

I made a [partition](#) and a recursive [quicksort](#) function to implement the quick sort algorithm.

The partition function takes the last element in the array as the pivot, places the pivot element at correct position in sorted array, and places elements smaller than pivot to the left of pivot and elements greater than pivot to the right of pivot. And it returns the partitioning index used as new pivots in recursive quicksort.

The quicksort sorts the left and right sub-arrays about the pivot.

I used a boolean variable `asc` to control the order of sorting.

Please look at my comments in the `ext1.rb` file for more detailed explanation.

Output:

My program is able to sort any type of array in ascending or descending order.

```
Unsorted:
[11, 1, 5, 17, 2, 3, 7]
Sorted in ascending order:
[1, 2, 3, 5, 7, 11, 17]
Sorted in descending order:
[17, 11, 7, 5, 3, 2, 1]

Unsorted:
["apples", "grapes", "bananas", "pears", "blueberries", "oranges", "mangos"]
Sorted in ascending order:
["apples", "bananas", "blueberries", "grapes", "mangos", "oranges", "pears"]
Sorted in descending order:
["pears", "oranges", "mangos", "grapes", "blueberries", "bananas", "apples"]
```

## Resources

<https://medium.com/@andrewsouthard1/quicksort-implementation-in-ruby-92de12470efd>