

### I.T 3 Search function

```
binary_search_ruby.rb x
1 class BinarySearch
2   attr_reader :array
3   def initialize()
4     @array = ['a', 'b', 'c', 'd', 'e']
5
6   end
7
8   def binary_search(item_to_find)
9     high = (@array.size) - 1
10    low = 0
11
12    while (low <= high)
13      mid = (high + low) / 2
14
15      if @array[mid] == item_to_find
16        return array[mid]
17      elsif @array[mid] < item_to_find
18        low = mid + 1
19      else
20        high = mid - 1
21      end
22    end
23    return nil
24  end
25 end
26
27
28 search = BinarySearch.new
29 p search.binary_search('b')
```

```
example_code — user@CODECLAN059 — -zsh — 80x24
../example_code      psql  +
[→ example_code git:(master) * ruby binary_search_ruby.rb
"b"
→ example_code git:(master) * ]
```

## I.T 4 Sort Function

```
own_sort.rb
* 1 require('pry')
* 2 class OwnSearch
* 3
* 4   def self.sorting(array)
* 5     return array.first if array.size <= 1
* 6     sorted_arr = Array.new()
* 7     #iterate through the given array and each iteration find the largest and smallest values
* 8     while array.size > 0
* 9       count = 0
*10       smallest = array[0]
*11       smallest_index = 0
*12       array.each do |item|
*13         smallest_index = count if item < smallest
*14         smallest = item if item < smallest
*15         count += 1
*16       end
*17       sorted_arr.push(array[smallest_index])
*18       array.delete_at(smallest_index)
*19     end
*20     return sorted_arr
*21   end
*22 end
*23
*24 p OwnSearch.sorting(['q','c','z','b', 's', 'd'])
*25 p OwnSearch.sorting([9,8,7,6,5,4,3,2,1])
```

```
pda — user@CODECLAN059 — -zsh — 80x26
..clan_work/pda      psql  +
[→ pda git:(master) × ruby example_code/week_3/own_sort.rb
["b", "c", "d", "q", "s", "z"]
[1, 2, 3, 4, 5, 6, 7, 8, 9]
→ pda git:(master) × █
```

## I.T 5 Use of an array in a program

```
my_array.rb — example_code
my_array.rb
1 def count_the_kittens(array_of_cats)
2   return array_of_cats.select{|feline| feline == "kitten"}.count
3 end
4
5 p count_the_kittens(["cat", "cat", "kitten", "kitten", "cat", "cat", "kitten"])
```

```
example_code — user@CODECLAN059 — ../example_code — zsh — 80x24
[→ example_code ruby my_array.rb
3
→ example_code █
```

## I.T 6 Use of a hash in a program

```
runner.rb
1 require_relative('person.rb')
2 require_relative('medic.rb')
3 require_relative('agent.rb')
4
5
6 person = Person.new("Chris", "Hunter")
7 medic = Medic.new("Steph", "Beattie")
8 agent = Agent.new("James", "Blonde")
9 hash_of_people = {
10   person1: person,
11   person2: medic,
12   person3: agent
13 }
14
15
16 hash_of_people.each{|num, person | p person.talk}
```

```
own_inheritance — user@CODECLAN059 — ..n_inheritance — -zsh — 80×24
[➔ own_inheritance ruby runner.rb
"Hi, I'm Chris Hunter"
"Hi, I'm Steph Beattie"
"The names Blonde, James Blonde"
➔ own_inheritance █
```



