

CS218 - Data Structures
FAST NUCES Peshawar Campus
Dr. Nauman (recluze.net)

October 21, 2019

1 Sorting

Raster images of the notebook 15-sorts.

Bubble Sort

```
In [3]: def bubble_sort(l):  
        n = len(l)  
        # print(n)  
  
        # Outer loop. Goes over the whole thing `n` times  
        # (because each time, one 'highest' will have moved to the end)  
        for i in range(n):  
  
            # try to bubble the highest one up  
            for j in range(0, (n-i)-1):  
  
                # compare pairs, move higher one up (the highest will always reach the end this way)
```

Selection Sort

```
In [ ]: def selection_sort(l):  
        n = len(l)  
  
        # for each element in the list (starting from left)  
        for i in range(n):  
            min_idx = i    # find the minimum ...  
  
            # .... in the *rest* of the list  
            for j in range(i+1, n):  
  
                if l[j] < l[min_idx]:  
                    min_idx = j  
  
            # swap the minimum with current element, now we have (sorted stuff till i)  
            l[i], l[min_idx] = l[min_idx], l[i]
```

```
In [ ]: l = [1, 2, 4, 1, 2, 5, 5, 6, 1, 110, 15]  
        selection_sort(l)  
        print(l)
```

Quick Sort

```
In [ ]: import random  
  
        def qsort(l, fst, lst):  
            if fst >= lst: return  
  
            i, j = fst, lst  
            pivot = l[random.randint(fst, lst)]
```


Sorting in Python

If you have a list of dictionaries -- each representing a student, for instance.

```
In [ ]: d = [
        { 'name': 'khalid', 'age': 5 },
        { 'name': 'usman', 'age': 7 },
        { 'name': 'ali', 'age': 12 },
        { 'name': 'farooq', 'age': 3 },
        ]
```

```
In [ ]: def d_less_than(a, b):
        return a['age'] < b['age']
```

```
sorted(d, key=d_less_than)
```

Sorting Objects of Custom Classes

```
In [ ]: class Student:
        def __init__(self, name, age):
            self.name = name
            self.age = age
        def __str__(self):
            return self.name + ': ' + str(self.age)
```

```
In [ ]: s1 = Student('Wajid', 5)
        s2 = Student('Usman', 7)
        s3 = Student('Ali', 3)

        s = [s1, s2, s3]
```