Lecture 16

Queries

In later section, remember to talk about raise!!!

1 Sorting

Both sections described selection sort, so we're going with that!

$$[25,\ 13,\ -100,\ -12,\ 60,\ 31,\ -76,\ 2]$$

- Find the minimum (select it), swap with index 0
- Repeat for spots $1,\ 2,\ \ldots,\ n-1$

Go through example above; write out code.

```
def selection_sort(lst):
    for index in range(len(lst) - 1):
        small_index = index_of_smallest(lst, index)

        swap(lst, index, small_index)


# Remind them again that functions can modify lists!
def swap(lst, i, j):
    tmp = lst[i]
    lst[i] = lst[j]
    lst[j] = tmp


def index_of_smallest(lst, start):
    min_index = start

    for index in range(start + 1, len(lst)):
        if lst[index] < lst[min_index]:
            min_index = index

    return min_index
```

2 PPM

Let's learn about ppm files! Specifically the P3 format of ppm files! The ppm P3 format is a plain text format for storing images. Let's take a look at making an image!

```
P3
3 1
255
255 0 0 # red
0 255 0 # green
0 0 255 # blue
```

Now let's make one that's a little bigger!

```
P3
3 2
255
255 0 0   # red
0 255 0   # green
0 0 255   # blue
255 255 0 # yellow
255 0 255 # magenta
0 255 255 # cyan
```

Mess with components per line, show them that it makes no difference/they'll need to be able to handle it. Then intro Program 6.

Truth Tables!

Start simple: p or q
Get harder: not ((not p or q) and (not q or r)) or (not p or r)

Methods?

Git?

Insertion Sort?

- Start at index 1, swap with lower one until it gets to the correct location (insert it).
- Each iteration, the first i elements of the list are sorted, each subsequent element is then inserted.

Go through example above. Write out code?

```
def insertion_sort(lst):
    for index in range(1, len(lst)):
        insert(lst, index, lst[index])


def insert(lst, location, to_insert):
    while location > 0 and to_insert < lst[location - 1]:
        lst[location] = lst[location - 1]
        location -= 1

    lst[location] = to_insert
```