part 2:

a)

part2_a.py

b)

```
def sort_arrays(a,b):
    total = [a,b]     ---O(1)
    for s in range(len(total)):--O(n)
       for i in range(len(total[s])):------O(n) * O(n)
         for j in range(0, len(total[s])-i-1): -----O(n) * O(n) * O(n)
           if total[s][j] > total[s][j+1]:   -----O(n) * O(n)* O(n)
             total[s][j],total[s][j+1]=total[s][j+1],              total[s][j] -----O(1) *
O(n) * O(n) * O(n)


    total.append([])   -----O(1)
    for i in range(len(total[1])):   -----O(n)
        total[2].append(total[0][total[1][i] - 1] )-----O(1)*O(n)
    return "Sorted A: " + str(total[2]) + "\n" + "Sorted B: " + str(total[1])
-----O(1)
a = [7, 3, 8, 21, 5, 11] -----O(1)
b = [3, 5, 1] -----O(1)
print(sort_arrays(a,b)) -----O(1)
```

```
total = 1 + n + n^2 + n^3 + n^3 + n^3 + 1 + n + n + 1 + 1 + 1 + 1
    = 6 + 3n + n^2 + 3n^3
    = O(n^3)
```

c)

part2_c.py

d)

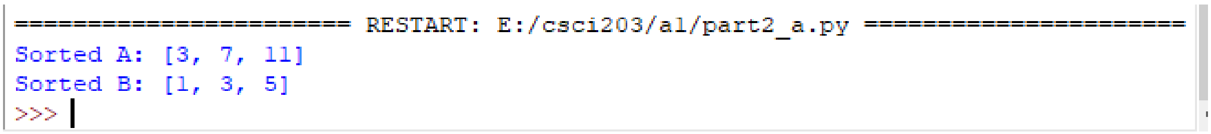
```
def sort_arrays(a,b):
    for i in range(len(a)):  ----O(n)
```

```
        for j in range(0, len(a)-i-1):  ----O(n)*O(n)
            if a[j] > a[j+1]:  ----O(1) * O(n)*O(n)
                a[j], a[j+1] = a[j+1], a[j] ----O(1) * O(n)*O(n)
    new_a = [] ----O(1)
    for i in range(len(b)): ----O(n)
        new_a.append(a[b[i]-1]) ----O(1) * O(n)
    return "Sorted A: " + str(new_a) + "\n"  ----O(1)
a = [7, 3, 8, 21, 5, 11] ----O(1)
b = [3, 5, 1]   ----O(1)
print(sort_arrays(a,b))  ----O(1)
```

total = n + n^2 + n^2 + n^2 + 1 + n + n + 1 + 1 + 1 + 1
      = 3n^2 + 3n + 5
      = O(n^2)

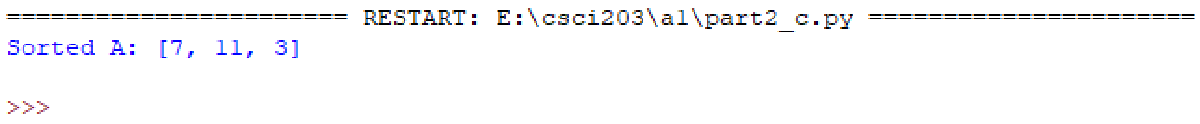
e)

Part 2a Screenshot



```
======================= RESTART: E:/csci203/a1/part2_a.py ======================
Sorted A: [3, 7, 11]
Sorted B: [1, 3, 5]
>>> 
```

Part 2c Screenshot



```
======================= RESTART: E:\csci203\a1\part2_c.py ======================
Sorted A: [7, 11, 3]

>>>
```