

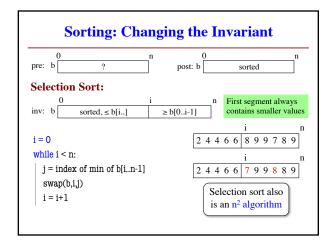
```
Insertion Sort: Moving into Position
i = 0
while i < n:
                                         2 4 4 6 6 7 5
  push_down(b,i)
  i = i+1
                                         2 4 4 6 6 5 7
def push_down(b, i):
  j = i
                                         2 4 4 6 5 6 7
  while j > 0:
                      swap shown in the
    if b[j-1] > b[j]:
                      lecture about lists
       swap(b,j-1,j)
                                         2 4 4 5 6 6 7
    j = j-1
```

```
Insertion Sort: Performance
def push_down(b, i):
                                  • b[0..i-1]: i elements
  """Push value at position i into
                                     Worst case:
  sorted position in b[0..i-1]"""
                                      ■ i = 0: 0 swaps
 j = i
                                      ■ i = 1: 1 swap
  while j > 0:
                                      ■ i = 2: 2 swaps
    if b[j-1] > b[j]:
                                    Pushdown is in a loop
      swap(b,j-1,j)
                                      Called for i in 0..n
    j = j-1
                                      ■ i swaps each time
                    Insertion sort is
                    an n<sup>2</sup> algorithm
   Total Swaps: 0 + 1 + 2 + 3 + ... (n-1) = (n-1)*n/2
```

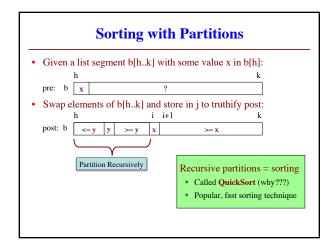
## Algorithm "Complexity" Given: a list of length n and a problem to solve Complexity: rough number of steps to solve worst case Suppose we can compute 1000 operations a second:

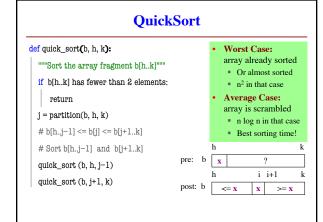
Complexity	n=10	n=100	n=1000
n	0.01 s	0.1 s	1 s
n log n	0.016 s	0.32 s	4.79 s
$n^2$	0.1 s	10 s	16.7 m
$n^3$	1 s	16.7 m	11.6 d
2 <sup>n</sup>	1 s	4x10 <sup>19</sup> y	3x10 <sup>290</sup> y

**Major Topic in 2110:** Beyond scope of this course



## 





## Final Word About Algorithms Algorithm: Step-by-step way to do something Not tied to specific language Implementation: An algorithm in a specific language Many times, not the "hard part" Higher Level Computer Science courses: We teach advanced algorithms (pictures) Implementation you learn on your own