CHAPTER 4

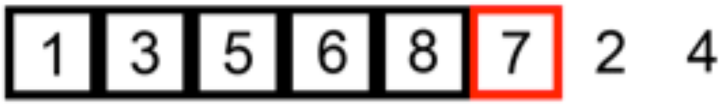
ALGORITHM ANALYSIS AND ASYMPTOTIC NOTATION

March 18, 2015

Lisa Yan

ANALYSE A SORTING ALGORITHM

Insertion sort



- ➔ grow a sorted list inside an unsorted list
- ➔ in each iteration
  - remove an element from the unsorted part
  - insert it into the correct position in the sorted part

see animation at: http://en.wikipedia.org/wiki/Insertion_sort

Insertion sort

6 5 3 1 8 7 2 4

n: size of A

```
def IS(A):
'''sort the elements in A in
   non-decreasing order'''

i = 1
while i < len(A):
    t = A[i]    # take red square out
    j = i
    while j > 0 and A[j-1] > t:
        A[j] = A[j-1]  # shift
        j = j - 1
    A[j] = t     # put red square in
    i = i + 1    # next element to be red-squared
```

i=1...n-1, that's n-1 iterations, + 1 final loop guard

j=i, …, 1, in worst case
that's i iterations,
+1 final loop guard,
total lines to run: 3i + 1

each iteration has (3i + 1) + 5 lines to execute

Insertion sort

$$1 + 1 + \sum_{i=1}^{n-1} [(3i + 1) + 5]$$

Line #1

Loop check

n: size of A

```
def IS(A):
'''sort the elements in A in
   non-decreasing order'''

1. i = 1
2. while i < len(A):
3.     t = A[i]    # take red square out
4.     j = i
5.     while j > 0 and A[j-1] > t:
6.         A[j] = A[j-1]  # shift
7.         j = j - 1
8.     A[j] = t      # put red square in
9.     i = i + 1     # next element to be red-squared
```

i=1...n-1, that's n-1 iterations, + 1 final loop check

j=i, …, 1, in worst case
that's i iterations,
+1 final loop guard,
total lines to run: 3i + 1

each iteration has (3i + 1) + 5 lines to execute

Insertion sort worst-case running time

$$W_{IS}(n) = 1 + 1 + \sum_{i=1}^{n-1} [(3i+1)+5]$$

$$= 2 + \sum_{i=1}^{n-1} (3i+6) = 2 + 6(n-1) + 3\sum_{i=1}^{n-1} i$$

$$= 6n - 4 + 3 \cdot \frac{n(n-1)}{2}$$

$$= \frac{3}{2}n^2 + \frac{9}{2}n - 4$$

Prove the worst case complexity of insertion sort is O(n^2)

$$W_{IS}(n) = \frac{3}{2}n^2 + \frac{9}{2}n - 4 \in \mathcal{O}(n^2)$$

$$\exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow W_{IS}(n) \leq cn^2$$

Proof:

Pick $c =$ **6**

Pick $B = 1$

Assume $n \in \mathbb{N}$

Assume $n \geq 1$

then $\frac{3}{2}n^2 + \frac{9}{2}n - 4 \leq$ **6** n^2

then $n \geq B \Rightarrow W_{IS}(n) \leq cn^2$

then $\exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow W_{IS}(n) \leq cn^2$

Prove $W_{IS}(n) = \frac{3}{2}n^2 + \frac{9}{2}n - 4 \in \Omega(n^2)$

$\exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow W_{IS}(n) \geq cn^2$

Proof:

Pick $c = \frac{3}{2}$

Pick $B = 1$

Assume $n \in \mathbb{N}$

Assume $n \geq 1$

then $\frac{3}{2}n^2 + \frac{9}{2}n - 4 \geq \frac{3}{2}n^2 + \frac{9}{2} \times 1 - 4 = \frac{3}{2}n^2 + \frac{1}{2}$

$\geq \frac{3}{2}n^2$

then $n \geq B \Rightarrow W_{IS}(n) \geq cn^2$

then $\exists c \in \mathbb{R}^+, \exists B \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq B \Rightarrow W_{IS}(n) \geq cn^2$

Complexity Analysis

The worst case time complexity of **insertion sort** is in **$O(n^2)$** and in **$\Omega(n^2)$**, i.e., it's in **$\Theta(n^2)$**

Summary

- we first derived the **exact form** of **WIS(n)**, then determined it's upper and lower bounds