Jordan Giebas
CSE 331

HW 2

Pr 4

Problem 4

(a)

Algo	Cost	# of times
Sum=0;	c_1	$\Theta(1)$
for $i=1$ to N do	c_2	n
Sum = Sum+i;	c_3	$\Theta(1)$

$T(n) = c_1\Theta(1) + c_2 \cdot n + c_3\Theta(1)$, let $c_i = 1 \ \forall i = 1,2,3$

$\Rightarrow$

$T(n) = 2\Theta(1) + n \Rightarrow \boxed{T(n) = \Theta(n)}$ (1 loop)

(b)

Sum=0;	c_1	$\Theta(1)$
for $i = 1$ to N do	c_2	N
for $j=1$ to N do	c_3	Σj
for $k=1$ to N do	c_4	Σk^2
Sum = Sum + 1	c_5	$\Theta(1)$

$T(n) = c_1\Theta(1) + c_2N + c_3\sum_1^n i + c_4\sum_1^n i^2 + c_5\Theta(1)$

let $c_i = 1 \ \forall i$

$\Rightarrow$

$T(n) = 2\Theta(1) + n + \frac{n(n+1)}{2} + \frac{n(n+1)(2n+1)}{6}$.

$\Rightarrow$

$\boxed{T(n) = \Theta(n^3)}$ (3 nested for loops)

Pr 3

(c)

Sum = 0;	c_1	$\theta(1)$
for i = 1 to N^2	c_2	$n = N^2$
for j = 1 to N	c_3	$\sum_{j=1} t_j$
Sum = Sum + 1	c_4	

$\Theta(n^2)$

Problem 3

(a)

		(b) Analysis
for i ← 1 to length[A]{	c_1	n
if (A[i] = val){	c_2	$\theta(1)$
return true}	c_3	$\theta(1)$
}		
return false (null)	c_4	$\theta(1)$

$\boxed{\text{pseudo-code}}$ let $c_i = 1 \; \forall i$

$\Rightarrow T(n) = n + 3\theta(1)$

$\rightarrow \boxed{T(n) = \Theta(n)}$

b. (a) Best case? ~~Suppose it's the first element you're looking for & it's the first in the list~~

Suppose the elm you're looking for is the first in the list, then $\theta(1)$.

(b) Worst Case?

Suppose the elm you're looking for is the last in the list, $\theta(n)$

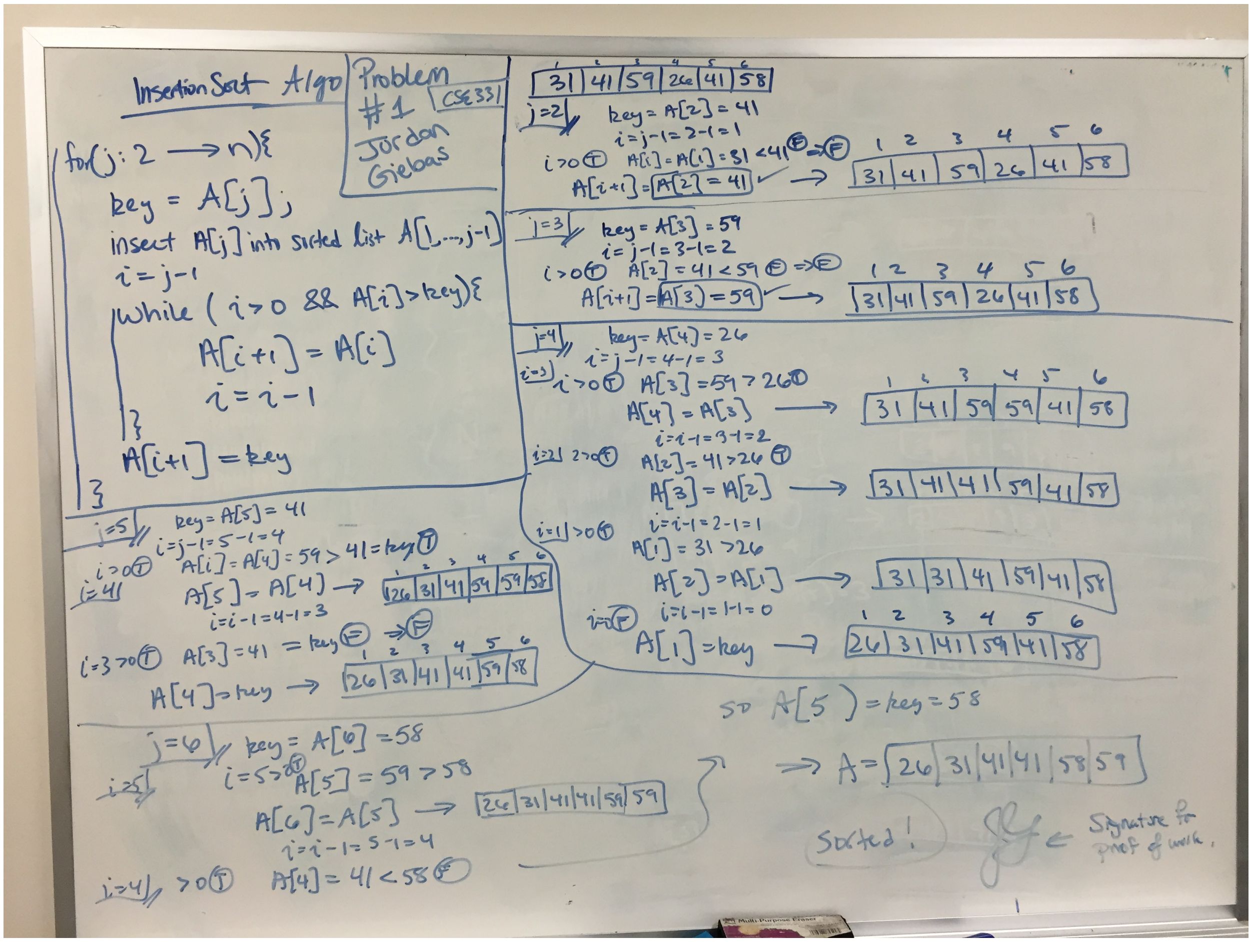
(c) Avg. Case?

n cases can occur, finding at the first, 2^{nd}, 3^{rd} ... n^{th} position. Take avg.

$$avg = \frac{1 + \dots + n}{n} = \frac{\sum_{i=1}^{n} i}{n} = \frac{\frac{n(n+1)}{2}}{n} = \frac{n+1}{2}$$

$= \theta(n)$.

Pr 1


Insertion Sort Algo
Problem #1
CSE331
Jordan Giebas
for(j: 2 → n){
key = A[j];
insert A[j] into sorted list A[1,...,j-1]
i = j-1
while (i > 0 && A[i] > key){
A[i+1] = A[i]
i = i-1
}
A[i+1] = key
}
31 | 41 | 59 | 26 | 41 | 58
j=2
key = A[2] = 41
i = j-1 = 2-1 = 1
i>0 T A[i] = A[1] = 31 < 41 F ⇒ F
A[i+1] = A[2] = 41
31 | 41 | 59 | 26 | 41 | 58
j=3
key = A[3] = 59
i = j-1 = 3-1 = 2
i>0 T A[2] = 41 < 59 F ⇒ F
A[i+1] = A[3] = 59
31 | 41 | 59 | 26 | 41 | 58
j=4
key = A[4] = 26
i = j-1 = 4-1 = 3
i=3 i>0 T A[3] = 59 > 26 T
A[4] = A[3]
31 | 41 | 59 | 59 | 41 | 58
i = i-1 = 3-1 = 2
i=2 2>0 T A[2] = 41 > 26 T
A[3] = A[2]
31 | 41 | 41 | 59 | 41 | 58
i = i-1 = 2-1 = 1
i=1 >0 T A[1] = 31 > 26
A[2] = A[1]
31 | 31 | 41 | 59 | 41 | 58
i=0 F i = i-1 = 1-1 = 0
A[1] = key
26 | 31 | 41 | 59 | 41 | 58
j=5
key = A[5] = 41
i = j-1 = 5-1 = 4
i>0 T A[i] = A[4] = 59 > 41 = key T
i=4
A[5] = A[4]
26 | 31 | 41 | 59 | 59 | 58
i = i-1 = 4-1 = 3
i=3 >0 T A[3] = 41 = key F ⇒ F
A[4] = key
26 | 31 | 41 | 41 | 59 | 58
j=6
key = A[6] = 58
i=5 i = 5 > 0 T
A[5] = 59 > 58
A[6] = A[5]
26 | 31 | 41 | 41 | 59 | 59
i = i-1 = 5-1 = 4
i=4 >0 T A[4] = 41 < 58 F
so A[5] = key = 58
A = 26 | 31 | 41 | 41 | 58 | 59
Sorted!
Signature for proof of work.