Lecture 2

Wednesday, January 13, 2021 11:21 AM

③ Time Complexity

```
INSERTION_SORT(A)
n = length(A);
for j = 2 : n
    temp=A(j);
    i = j - 1;
    while ((i > 0) && (x(i) > temp))
        A(i+1) = A(i);
        i = i - 1;
    A(i + 1) = temp;
```

Cost	Count
c_1	1
c_2	n
c_3	$n-1$
c_4	$n-1$
c_5	t_j
c_6	$t_j - 1$
c_7	$t_j - 1$
c_8	$n-1$

; t_j: random Var.

$\min(t_j) = 1$; $\max(t_j) = j$

We can write $T(n)$ as

$$T(n) = C' + B'n + D' \sum_{j=2}^{n} t_j$$

C', B', D' are Constants

Two Cases: [1] $A(1..n)$ is already sorted

$t_j = 1$ for every j

$$T(n) = C' + B'n + D' [1 + 1 + \dots + 1]$$

($n-1$ terms)

$$T(n) = \alpha + \beta n \Leftarrow \text{Linear.} \quad -(A)$$

[1] Worst Case "A is sorted in the reverse order"

$$T(n) = C' + B'n + D' [2 + 3 + 4 + \dots + n]$$

$$T(n) = C + Dn + D[2+3+4+\cdots+n]$$

(arithmetic Seq.)

$$= C' + Bn + D'\left(\frac{2+n}{2}\right)(n-1)$$

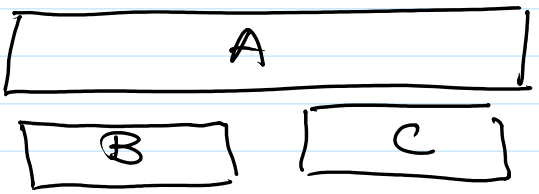
$$= \alpha + \beta n + \gamma(n^2) \quad \text{— (B)}$$

In this case running time is Quadratic

Merge Sort

Basic Idea

1. Divide Array A into Two Sub_array



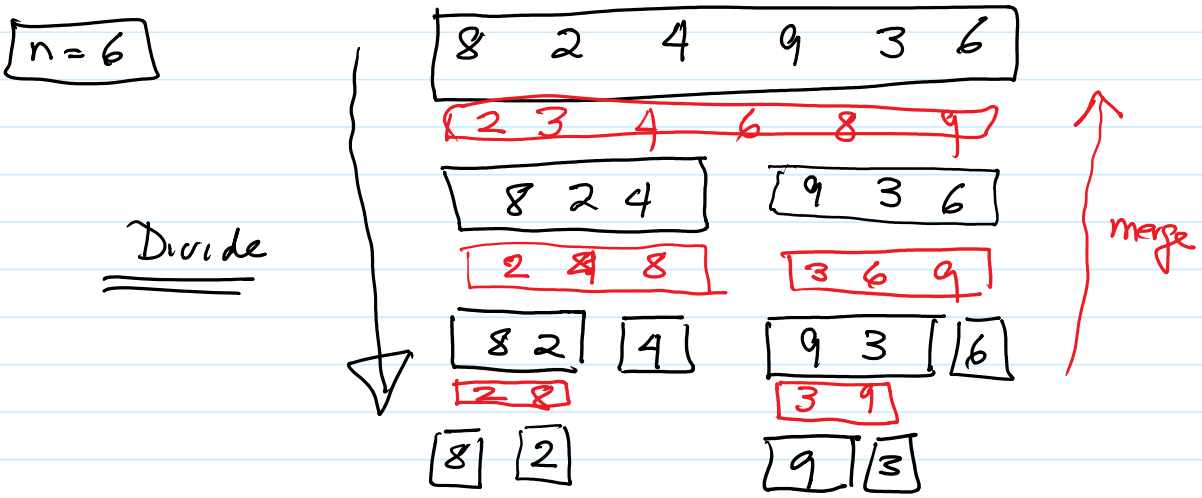
Divide AND Conquer technique

2 Conquer — which means to sort B and C

3. Combine or Merge the Conquered B and C

Numerical Example

Numerical Example



Alg. Merge (A[1 -- n])

If n == 1 ⇒ Done

else

Merge (A[1 -- $\lceil \frac{n}{2} \rceil$])

Merge (A[$\lceil \frac{n}{2} \rceil$ + 1 , n])

Combine the two Sublists

MATLAB DEMO

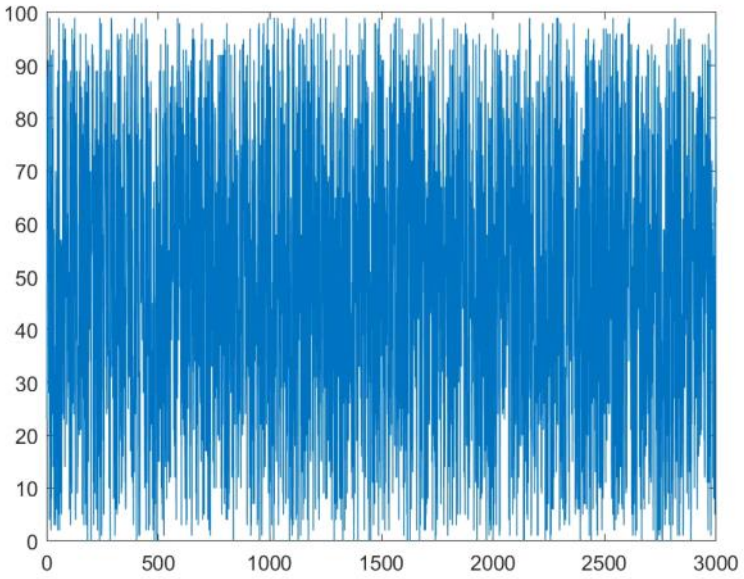
```
function y = INSERTION_SORT(x)
n = length(x);
for j = 2 : n
   temp=x(j);
   i = j - 1;
   while ((i > 0) && (x(i) > temp))
      x(i+1) = x(i);
      i = i - 1;
   end
   x(i + 1) = temp;
end
y = x;
end
```

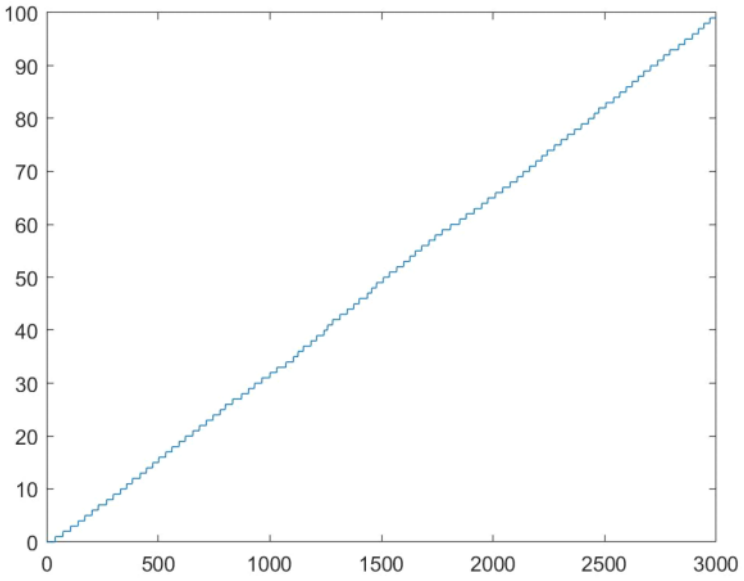
```
==========================================================================
>> for j=1 : 3000 x(j)=floor(rand()*100);end;
>> tic;y=INSERTION_SORT(x);t=toc;
>> t

t =

   0.1198

>> plot(x)
>> figure;plot(y)
>>
==========================================================================
```


100
90
80
70
60
50
40
30
20
10
0
0
500
1000
1500
2000
2500
3000


100
90
80
70
60
50
40
30
20
10
0
0
500
1000
1500
2000
2500
3000