CMPS 10 10-17-08

EX. CONSIDER 4 ALGORITHMS:

A: $1000n$ (BASIC OPS.) } $\Theta(n)$

B: $10n^2$ " }

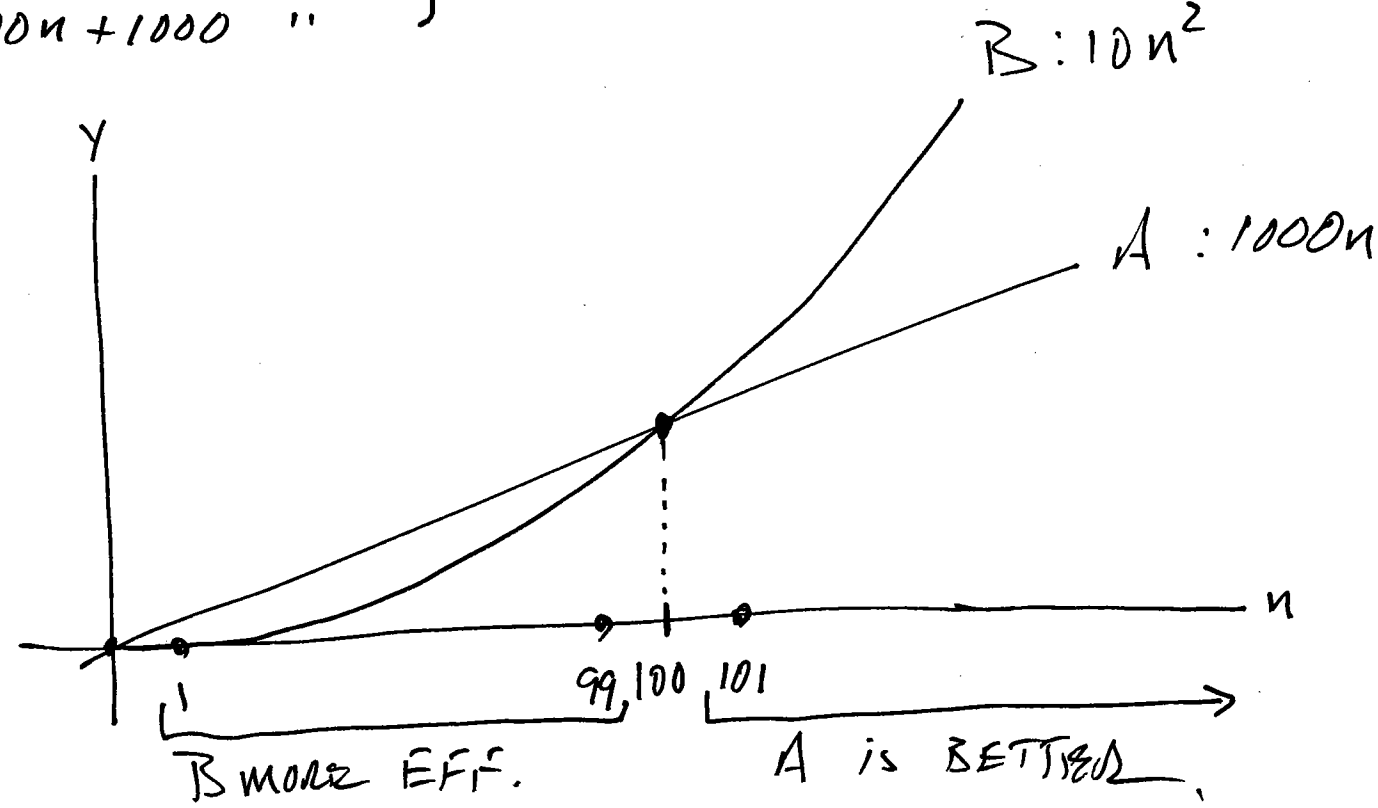
C: n^2 " } $\Theta(n^2)$

D: $n^2 + 100n + 1000$ " }

COMPARE A & B

$1000n = 10n^2$

$100 = n$



COMPUTER B ! C: CAN EQUALIZE BY RUNNING B ON A MACHINE WHICH IS 10 TIMES FASTER

COMPUTER C ! D: LOWER ORDER TERMS ARE A SMALL % OF TOTAL COST FOR LARGE n

$$\frac{n^2+100n+1000}{n^2} = 1 + \frac{100}{n} + \frac{1000}{n^2} \to 1$$

$\frac{100}{n} \to 0$ $\quad \frac{1000}{n^2} \to 0$

Ex (Possible Quiz Question)

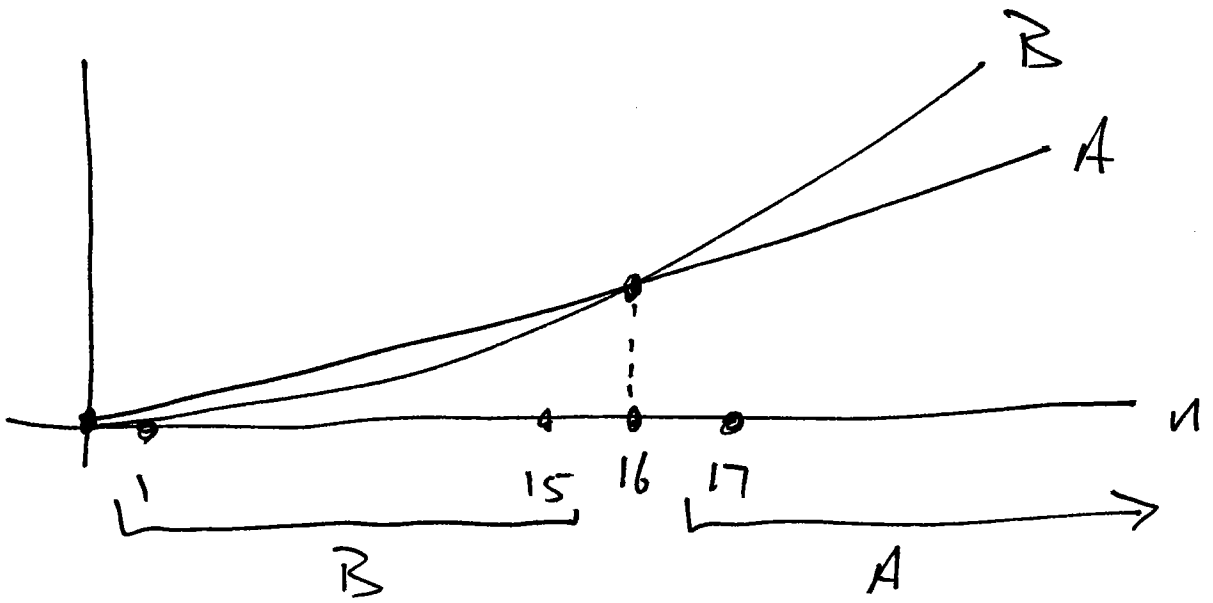
Consider Algorithms A & B

A: $12n\sqrt{n}$ basic ops $\Theta(n^{3/2})$

B: $3n^2$ " " $\Theta(n^2)$

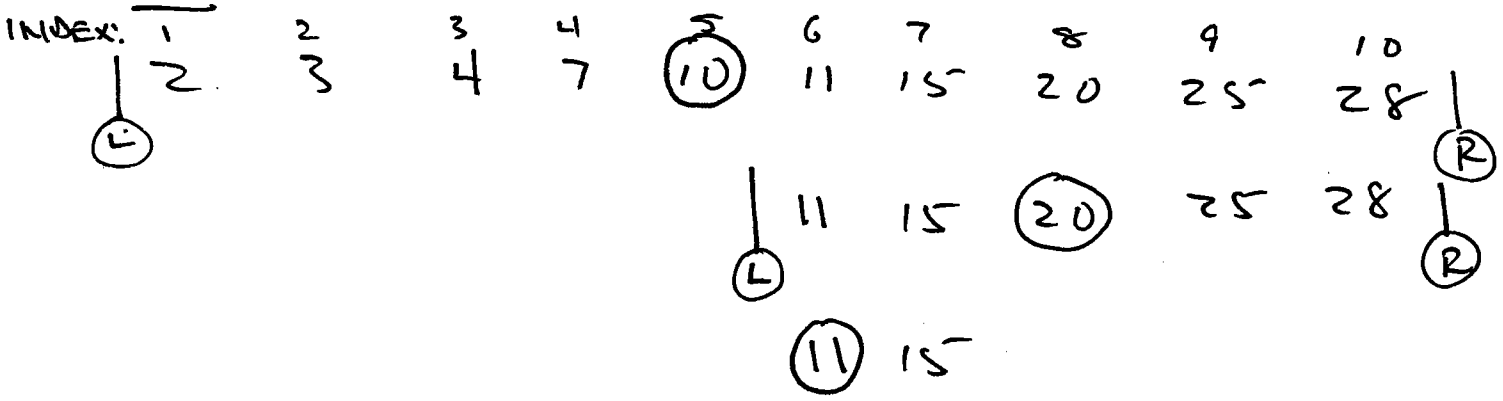
For which n are A & B equally efficient?

$$12n\sqrt{n} = 3n^2 \Rightarrow 4 = \sqrt{n} \Rightarrow \boxed{n = 16}$$

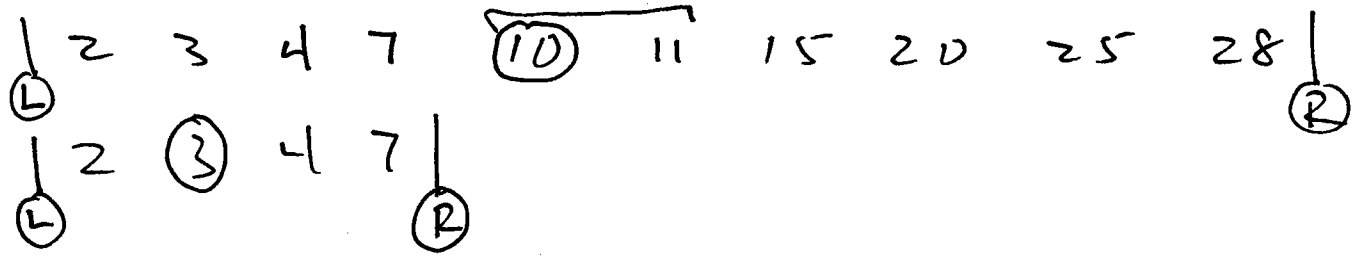


Binary Search

Ex. $n = 10$, $T = 11$ #comp = 3



Ex. $T = 3$ #comp = 2



Ex. same list : $T = 7 \Rightarrow$ 4 comparisons

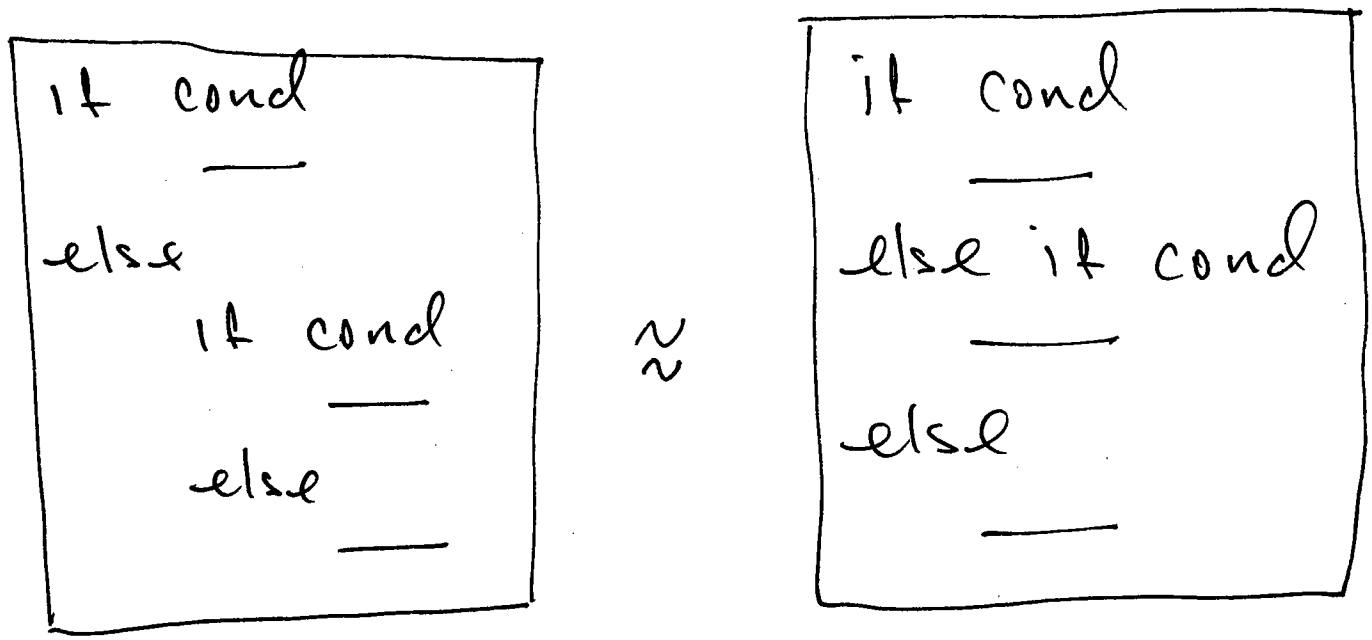
Ex. " " : $T = 8 \Rightarrow$ 4 comparisons

Binary Search

```
1.) L ← 1
2.) R ← n
3.) found ← false
4.) while L ≤ R and not found
5.)     m ← ⌊(L+R)/2⌋
6.)     if target = a_m
7.)         found ← true
8.)     else if target < a_m
9.)         R ← m-1
10.)    else
11.)        L ← m+1
12.) if not found
13.)     m ← 0
14.) Print m
15.) stop
```

Def'n:
$\lfloor x \rfloor$ = floor of x
i.e. integer part of x.

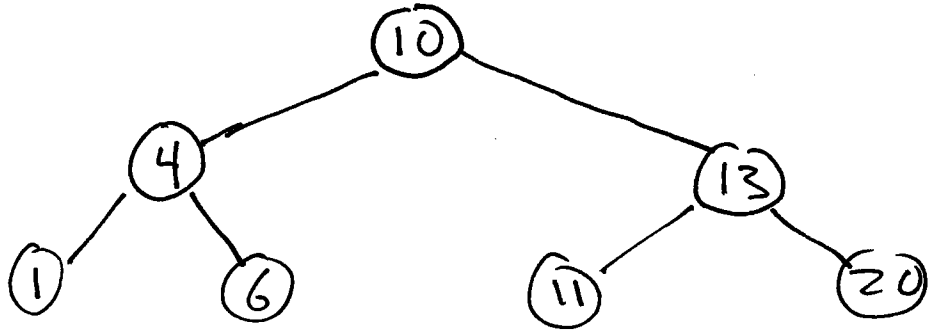
NOTATION



Binary Search Trees

Ex. n = 7

Index:	1	2	3	4	5	6	7
	1	4	6	10	11	13	20



Comp
1
2
3

Worst case # of comp = 3