

Intro CP 6/10/26

1) Iterative back tests

2) Maps

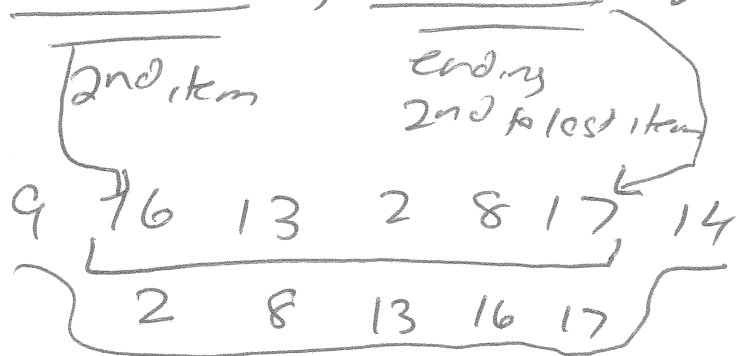
$$\text{ceiling}\left(\frac{x}{y}\right) = (x+y-1)/y;$$

$$\left\lceil \frac{n+c}{16} \right\rceil$$

$$\underline{\underline{(n+c+15)/16}}$$

if and do works.

sort(arr.begin()+1, arr.end()-1)



```
for (int x = ... ) {  
    int ans = 0;  
    int px = 1;
```

```
    for (int i = 0; i < f.size(); i++) {  
        ans = ans + f[i] * px;  
        px = px * x;  
    }
```

Store poly $3x^4 + 2x^3 - 6x^2 + 5x + 2$

0	1	2	3	4
3	2	-6	5	2

```
int ans = 0;
for (int i = 0; i < n; i++)
    ans = ans * x + f[i];
```

ans 0

ans $0x + 3$

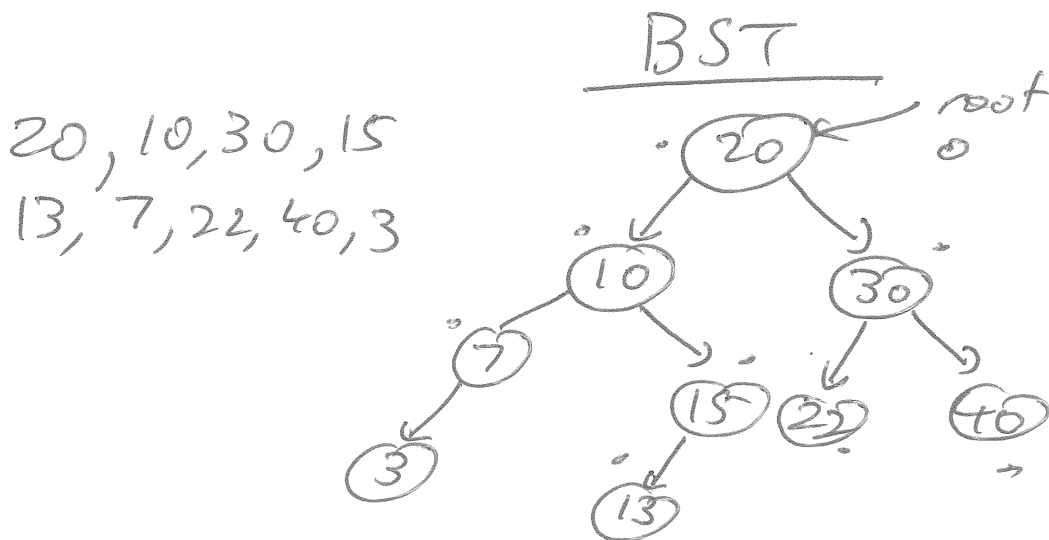
(Horner's Method)

ans $3x + 2$

ans $3x^2 + 2x - 6$

ans $3x^3 + 2x^2 - 6x + 5$

ans $3x^4 + 2x^3 - 6x^2 + 5x + 2$



20, 10, 30, 15
13, 7, 22, 40, 3

20 → 0
10 → 0+1 = 1
30 → 1+1 = 2
15 → 2+2 = 4

13 → 4+3 = 7
7 → 7+2 = 9
22 → 9+2 = 11
40 → 11+2 = 13
3 → 13+3 = 16

Depth
0

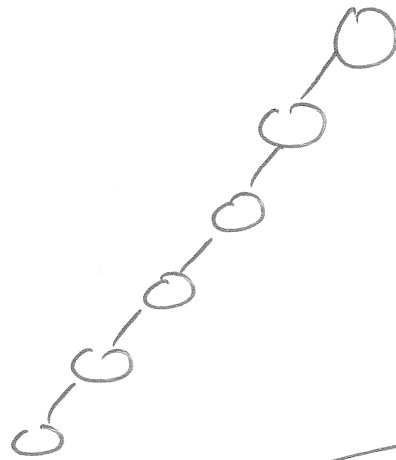
etc. 1

2

3

Case that runs a simulation too slow

300,000, 299,999, . . . 1



Can't actually
simulate!!!
too many steps
and answer can
be as large as

$$\frac{299,999 \times 300,000}{2}$$

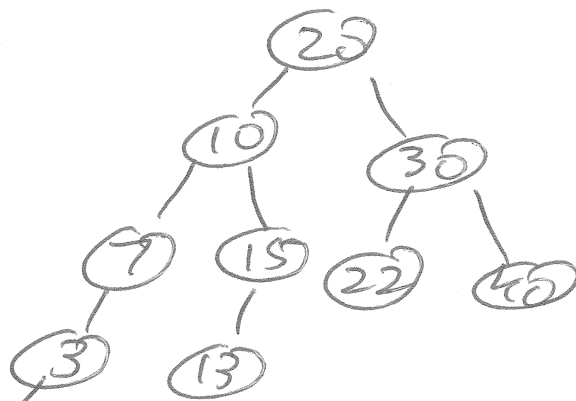
3 → 3

30 → 1

22 → 2

27 → 3

1 → 4



Inserting

22 ← 27 → 30

Inserting

X ← 1 → 3

Into ① we need depth of the 2 closest
nodes already in tree by value (one smaller,
one larger) → closest in each direction

Keep a map of node value to
depth. (could use a vector because
node values are 1 → N)