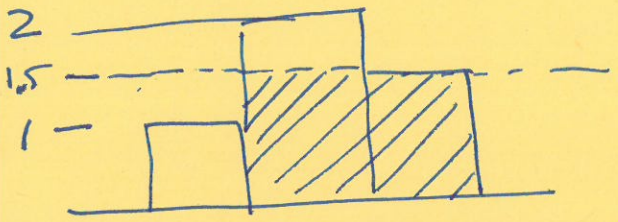
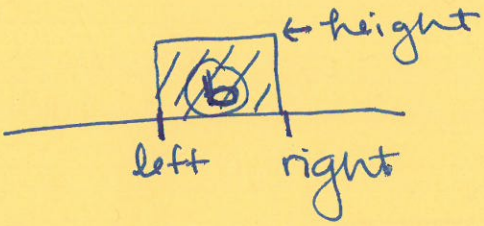
7 Oct 2019

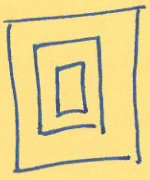


goal: find a axis aligned rectangle with max area.

Given: S = set of buildings (assume input is sorted)

$b \in S$ is defined by

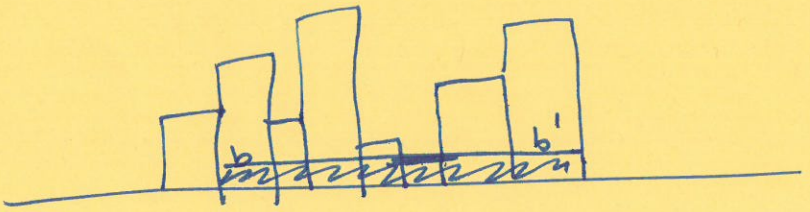


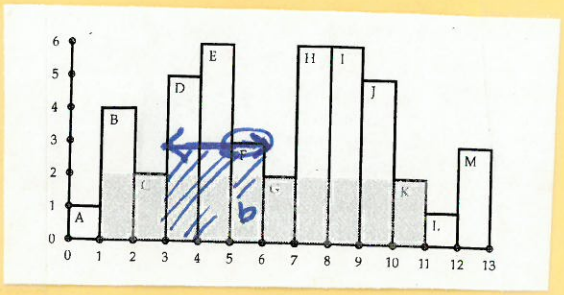


Try 1: $\Theta(n^3)$

```
For each building b ∈ S
    choose left
    for each building b' ∈ S
        choose right.
        for each building btwn b, b'
            find min hight
        end for
    end for
end for
```

"all combinations"





Try 2: $\Theta(n^2)$ – height first approach

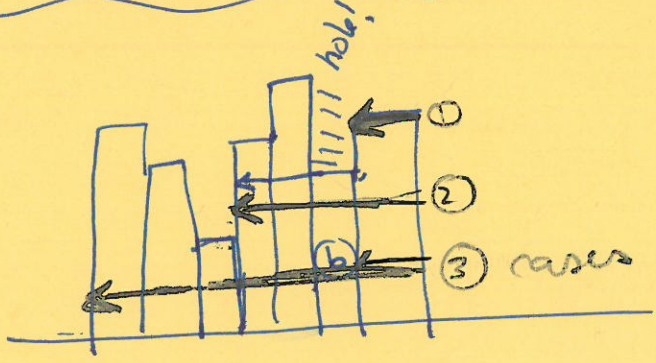
repeats $\Theta(n)$ times

for $b \in B$, in order

- $\Theta(1)$ choose the height
- $O(n)$ look left as far until lower found
- $O(n)$ look right as far until lower found

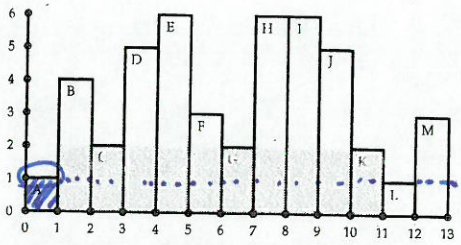
end for

$\Theta(n^2)$ operations



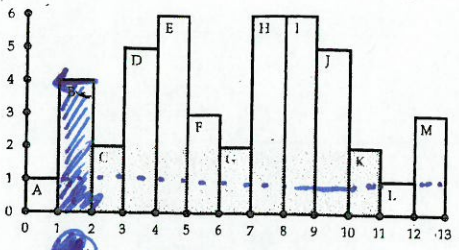
instead: let's use a stack & carefully count.

(1)



in progress: A

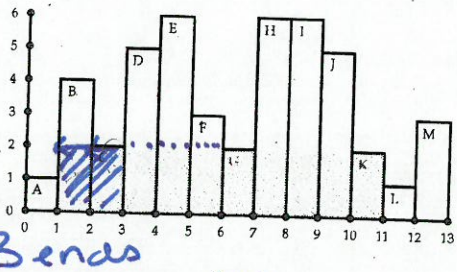
(2)



~~B ends~~

in progress: A, B

(3)

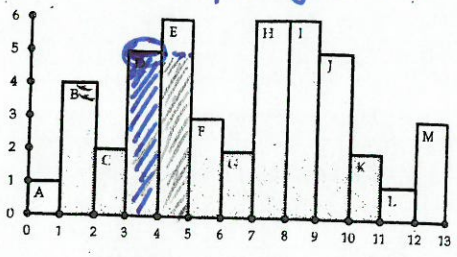


B ends

in progress: A, C

Best: B = 1×4 = 4 units

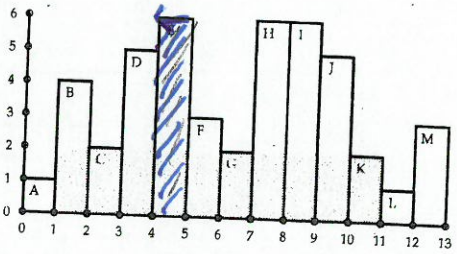
(4)



in progress: A, C, D,

Best: B = 4 units

(5)

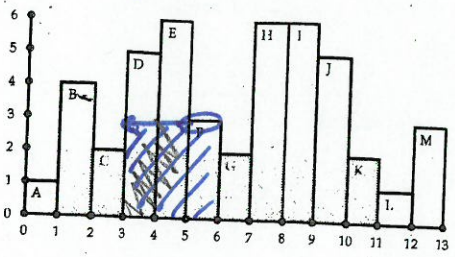


~~E ends + D ends~~

in progress: A, C, D, E

Best: B = 1×4 = 4 units

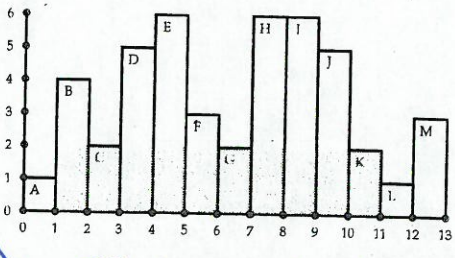
(6) now we see F

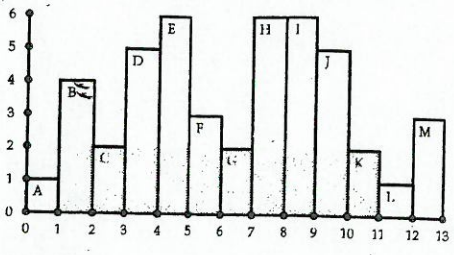


→ E ends, D ends

in progress A, C, F

Best: D = 5×2 = 10 units





Think of this as a stack.

charging ops:

c_i = actual cost

$\hat{c}_i$ = amortized cost

$$\sum \hat{c}_i \geq \sum c_i$$