stdint.h

existence is platform dependent

			range	200
uint64_t	unsigned 64-bit	int	0 to $2^{64}-1$	unsigned long int
int32_t	signed 32-bit	int	-2^{31} to $2^{31}-1$	

Stamps

1¢, 23¢, 37¢ stamps (unlimited supply of each)

Make 50¢ using fewest possible stamps $50 = 23 + 23 + 1 + 1 + 1 + 1$ (6 stamps)

In general: if $v_1, v_2, \ldots, v_k$ is shortest list with $v_i \in \{1, 23, 37\}$ and $\sum v_i = n$

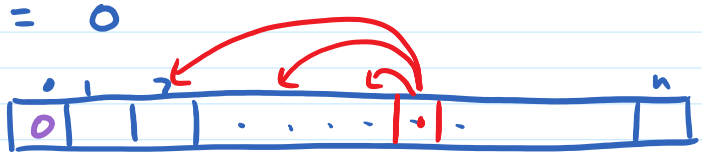
then $v_1, \ldots, v_{k-1}$ is shortest list with sum $n - v_k$

if it weren't — if $u_1, \ldots, u_\ell$ is better ($\ell < k-1$) way to make $n - v_k$ cents then $u_1, \ldots, u_\ell, v_k$ is better than best way to make n cents

optimal substructure

$$\underset{n>0}{num(n)} = 1 + \underset{n-denom \geq 0}{\min}\left(num(n-37), num(n-23), num(n-1) \right)$$

$$num(0) = 0$$



How many possible states in 3×4 Chomp? (h = rows, w = cols)

0000
1000
⋮
3333

does this depend on # of states in 3×3 Chomp?

2220
2221
2222

2210
2211
~~2212~~

1110
1111

3330
3331
3332
3333

2000

for h×w Chomp # states (with anything in rightmost col) = # h×(w−1) states that end in 0 · 1
+ (# h×(w−1) states that end in 1) · 2 (can add 0 or 1 in last col)
+ (# h×(w−1) states that end in 2) · 3 (can add 0,1, or 2 in last col)
+
⋮
+ (# h×(w−1) states that end in h) · (h+1)

So need to know # h×i states that end in k

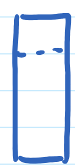
count(i,k) = # h×i Chomp states w/ k in rightmost column

= # h×(i−1) states w/ k in rightmost
h×(i−1) states w/ k+1 in rightmost
⋮
h×(i−1) states w/ h in rightmost

(can add another column of k to right of these)

= count(i−1, k) + count(i−1, k+1) + … + count(i−1, r)

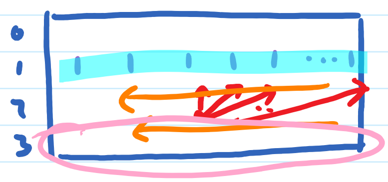
$$= \sum_{j=k}^{h} \text{count}(i-1, j)$$



count(1,k) = 1

count[w+1−1][h+1] (0…c−1, ~~0~~…c; 0…r)

add −1 so indices start @ 0 instead of 1



ans = sum of last row

init count[1−1][r] to 1 for r in 0…h

for c = 2 to w
 for r = 0 to h
  count[c−1][r] = $\Sigma_{j=r}^{h}$ count[c−1−1][j]

top-down, right-to-left

$O(wh^2)$ [can you improve to $O(wh)$?]

[can you improve to O(wh)?]

$count[c-1][r] - \sum_{j \geq r} count[c-1-1][j]$

[return sum of last row

(or extend table one more row and return 1st entry in that row, which =)

Image Resizing by Seam Carving

energy

	0	1	2	3	4	5
0	1	1	1	4	1	2
1	3	6	5	1	0	3
2	2	4	3	2	2	0
3	0	2	1	6	5	3
4	1	1	0	2	1	0

want seam with

Bellman-Ford

Let $v_0, v_1, \ldots, v_k$ be min-cost path $s \leadsto u$
($v_0 = s$, $v_k = u$)

Then $s, \ldots, v_{k-1}$ is

Let $d[u, k]$ = cost of min-cost path $s \leadsto u$ using $\leq k$ edges

Then