Greed Algorithms
- Very intuitive
- For some algorithms, less optimal
- You're able to reduce the problem size
  \( \rightarrow \) substructure optimality \( \Rightarrow \) induction

Knapsack Problem (special case)
Input: \( W_1, W_2, W_3, \ldots, W_n > k \)
  (objects)
Output: \( S \subseteq \{1, 2, \ldots, n\} \)
Want to output a subset such that \( \sum_{i \in S} W_i = k \)
\( \Rightarrow \) items aren't divisible

Encryption (which really advanced computer algos)

```
Sender \( \xleftarrow{\$} \) E \( \rightarrow \) Receiver
MSG \( \xrightarrow{E} \) MSG
E = ENC(MSG) \( \Rightarrow \) M = DEC(E)
```

...But if the sender and receiver haven't talked yet, how can they share a secret, private language?
\( \Rightarrow \) Public Key Encryption Scheme

"Yellow Book Encryption Scheme"
RSA is a variation of this

Merkle-Hellman (Stanford)

\[ \alpha_1, \alpha_2, \ldots, \alpha_n \quad \alpha_i = 1 \text{ iff } n \in S \]

\[ \alpha_i = 0 \]

\[ m = (m_1, m_2, \ldots, m_n) \]

Encryption

1) Choose \((w_1, w_2, \ldots, w_n)\) such that for all \(i\)

\[ w_i > \sum_{j=1}^{i-1} w_j \] (superincreasing)

\[ \text{Next number > sum of previous } 2 \text{ numbers} \]

\((2, 7, 11, 21, 42, 89, 180, 354) \quad \mathcal{E} = 706\)

2) Choose large random \(q\) such that \(q > \sum_{i=1}^{n} w_i\)

\[ q = 881 \]
3) Choose a large random \( r \) such that
\[
\text{gcd}(r, q) = 1
\]
\( r = 588 \)

4) Encryption key \((p_1, p_2, \ldots, p_n)\) such that
\[ p_i = r w_i \mod q \]

\( \Rightarrow \) this makes it seem random

\((295, 592, 201, 14, 28, 353, 120, 236)\)

\( \Rightarrow \) yellow page entry

5) Publish \((p_1, p_2, \ldots, p_n)\)

\( \Rightarrow \) public

Keep \((w_1, w_2, \ldots, w_n), q, r\)

\( \Rightarrow \) private

Encryption:
- message = 01100001
- pairwise multiply w/ yellow page entry
  \[ 592 + 301 + 236 = 1129 \] (ciphertext)

Decoding:
- just a knapsack problem
  ... but knapsack problem's hard ...

One-way-ness

\( \Rightarrow \) easier to encode, seemingly hard to decode

Knapsack problem for superincreasing sequences:

\( \Rightarrow \) greedy
\( w = (2, 7, 11, 21, 42, 89, 190, 354) \) \( k = 372 \)

\[
\begin{array}{cccccccc}
2 & 7 & 11 & 21 & 42 & 89 & 190 & 354 \\
\downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
\end{array}
\]

\( 372 - 354 = 18 \)

\( 18 - 11 = 7 \)

\( 7 - 7 = 0 \)

-Take the largest one you can

-Where did 372 come from?

\( q = 881 \) \( r = 588 \) \( s = 442 \)

\( \text{gcd}(q, r) = 1 \) \( \exists \) \( s, s \cdot r = 1 \mod q \)

\( 1129 \cdot 442 \mod 881 = 372 \)

This all failed after shortest vector problem was solved (unrelated problem)

Minimum spanning tree

- SEA
  - 900
  - 850
  - 940
  - 450
  - 1100

- DEN
  - 900
  - 1000
  - 1100
  - 1600

- CHI
  - 1000

- NY
  - 700
  - 1400

- DAS
  - 1600

- DC
  - 1100
  - 1550

- MIA
  - 1000
GIVEN THIS MAP, FIND A TREE CONNECTING ALL CITIES WITH THE SHORTEST TOTAL LENGTH

- DIVIDE MAP INTO 2 PARTS, THEN THE SHORTEST DISTANCE BETWEEN 1st & 2nd SETS BELONGS TO THE TREE

ALGORITHM 1
1) ORDER CITIES PAIRWISE BY DISTANCE
2) CONNECT SHORTEST
3) CONNECT NEXT SHORTEST UNTIL IT INDUCES A CYCLE
   → THEN THROW AWAY THE LINK THAT CREATES THE CYCLE
4) CONTINUE UNTIL THE COUNTRY IS CONNECTED

ALGORITHM 2
1) PICK A STARTING CITY AND CONNECT ITS CLOSEST NEIGHBOR
2) CONNECT THE NEXT CLOSEST CITY TO THE GROUP
3) THAT CITY IS NOW PART OF THE GROUP
4) REPEAT STEPS 2-3 UNTIL ALL CITIES ARE CONNECTED