Greedy Algorithm
Yang Wang

Greedy for Optimality

- A [Greedy Algorithm] is any algorithm that makes the locally optimal choice at each stage with the hope of finding the global optimum.

- [Optimization Problem] is a type of problem in which you want to find, not just a solution, but the best solution.

Specifically...

- A greedy algorithm works in phases.
- At each phase:
  - You take the best you can get right now, without regard for future consequences
  - You hope that by choosing a local optimum at each step, you will end up at a global optimum.

Huffman Encoding

- Greedy Choice: always pick the two smallest numbers to combine

- Average bits/char: 
  \[
  0.22\times2 + 0.12\times3 + 0.24\times2 + 0.06\times4 + 0.27\times2 + 0.09\times4 = 2.42
  \]
Bad News

- Short Sighted & Non-recoverable
- Local Optima != Global Optima

![Diagram](image)

When Greedy = Optimality?

- **[Greedy Choice Property]** We can make whatever choice seems best at the moment and then solve the sub-problems that arise later.
- **[Optimal Substructure]** An optimal solution to the problem contains optimal solutions to the sub-problems.
- **Proof of Optimality: Induction**
- Ex. On the Board

Also When..

- Matroids
- Greedoids

Heroes of Today

- David Huffman
Connecting Wires

- There are \( n \) white dots and \( n \) black dots, equally spaced, in a line.
- You want to connect each white dot with some one black dot, with a minimum total length of “wire”.
- Example:

![Diagram of connecting wires]

- Total wire length above is \( 1 + 1 + 1 + 5 = 8 \)
- Do you see a greedy algorithm for doing this?
- Does the algorithm guarantee an optimal solution?
  - Can you prove it?
  - Can you find a counterexample?

Collecting Coins

- A checkerboard has a certain number of coins on it.
- A robot starts in the upper-left corner, and walks to the bottom left-hand corner.
  - The robot can only move in two directions: right and down.
  - The robot collects coins as it goes.
- You want to collect all the coins using the minimum number of robots.
- Example:

![Diagram of collecting coins]

- Do you see a greedy algorithm for doing this?
- Does the algorithm guarantee an optimal solution?
  - Can you prove it?
  - Can you find a counterexample?