

# Greedy Algorithms: Review

Michael Levin

Department of Computer Science and Engineering  
University of California, San Diego

# Main Ingredients

- Safe choice
- Prove safety
- Solve subproblem
- Estimate running time

# Safe Choices

- Put max digit first
- Choose patient with min treatment time
- Cover leftmost point
- Take item with maximum value per unit of weight

# Optimization

- Assume everything is somehow sorted
- Which sort order is convenient?
- Greedy move can be faster after sorting

# Safe choice

- A choice is called **safe** if there is an optimal solution consistent with this first choice

# Safe choice

- A choice is called **safe** if there is an optimal solution consistent with this first choice
- Not all first choices are safe

# Safe choice

- A choice is called **safe** if there is an optimal solution consistent with this first choice
- Not all first choices are safe
- Greedy choices are often unsafe

# General Strategy

Problem

# General Strategy

Problem  $\xrightarrow{\text{greedy choice}}$

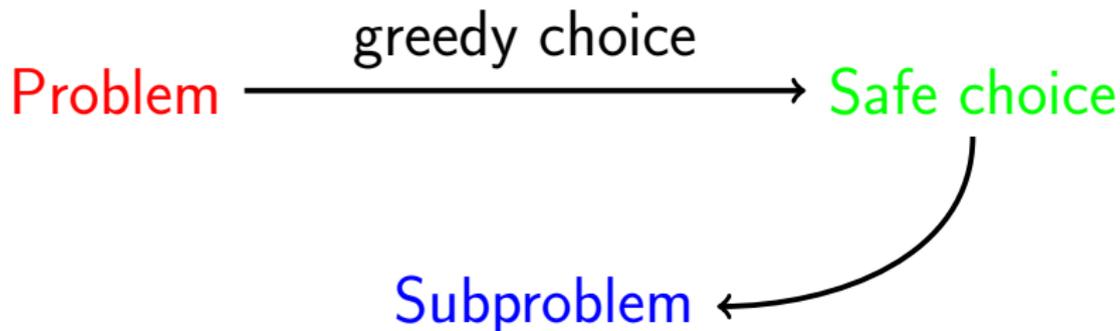
- Make a greedy choice

# General Strategy

Problem  $\xrightarrow{\text{greedy choice}}$  Safe choice

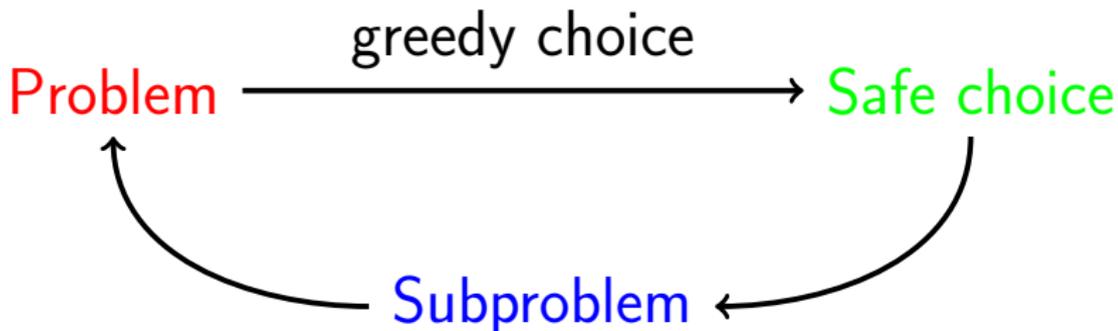
- Make a greedy choice
- **Prove** that it is a **safe choice**

# General Strategy



- Make a greedy choice
- **Prove** that it is a **safe choice**
- Reduce to a **subproblem**

# General Strategy



- Make a greedy choice
- **Prove** that it is a **safe choice**
- Reduce to a **subproblem**
- Solve the **subproblem**