Greedy

Greedy is good

beOI Training

OLYMPIADE BELGE D'INFORMATIQUE
BELGISCHE INFORMATICA-OLYMPIADE

Table of Contents

Greedy traits

Example problems

General remarks

”The point is, ladies and gentleman, that ’greed’, for lack of a better word, is good.”
Gordon Gecko, Wall Street

Traits of a greedy person

A greedy person

- Doesn't care about the future
- Doesn't dwell on the past
- Looks only at the present situation
- Takes the biggest/best thing currently available

Traits of a greedy algorithm

A greedy algorithm

- Makes the locally optimal choice at any state.
- Doesn't know anything about a future state.
- Doesn't go back for fixing mistakes.

Table of Contents

Greedy traits

Example problems

General remarks

A shortest path algorithm

A shortest path algorithm

At every point take the shortest edge and go from there, until you get to the destination.

A shortest path algorithm

At every point take the shortest edge and go from there, until you get to the destination.

Will this work? Counterexample?

A shortest path algorithm

At every point take the shortest edge and go from there, until you get to the destination.

Will this work? Counterexample?

No! This is not an algorithm, but a **heuristic** (use Dijkstra)

Coin change

You have a given set of coin types (ex: $\{25, 10, 5, 1\}$)
We have an unlimited amount of coins.
How can we give a certain amount of money with the least amount of coins?
Example: Give 42 cents back

Does the greedy algorithm work for every coin set?
Counterexample

Try making 6 cents with 4,3,1

Does it ever work?

Does it ever work?

... seems like it doesn't

Does it ever work?

... seems like it doesn’t

But sometimes it does!

Interval scheduling

A set of activities, each with a starting and ending time.
How can we schedule the most number of activities?

Let's try some ideas:

1. Earliest starting time?

Interval scheduling

A set of activities, each with a starting and ending time.
How can we schedule the most number of activities?

Let's try some ideas:

1. Earliest starting time? No!

Interval scheduling

A set of activities, each with a starting and ending time.
How can we schedule the most number of activities?

Let's try some ideas:

1. Earliest starting time? No!
2. Shortest interval?

Interval scheduling

A set of activities, each with a starting and ending time.
How can we schedule the most number of activities?

Let's try some ideas:

1. Earliest starting time? No!
2. Shortest interval? No!

Interval scheduling

A set of activities, each with a starting and ending time.
How can we schedule the most number of activities?

Let's try some ideas:

1. Earliest starting time? No!
2. Shortest interval? No!
3. Earliest ending time?

Interval scheduling

A set of activities, each with a starting and ending time.
How can we schedule the most number of activities?

Let's try some ideas:

1. Earliest starting time? No!
2. Shortest interval? No!
3. Earliest ending time? Yes!

Load balancing

Certain number of containers C.
Certain number of items S with a certain mass M_i.
$1 \leq S \leq 2C$
Minimize inbalance:

$$A = \frac{\sum_{j=1}^{S} M_j}{C}, Imbalance = \sum_{i=1}^{C} |X_i - A|$$

where X_i is the total mass in chamber i

Load balancing

Any idea?

Load balancing

Any idea?

Here's a hint: make sure there are exactly $2C$ items by adding dummy elements.

Load balancing

Any idea?

Here's a hint: make sure there are exactly $2C$ items by adding dummy elements.

Sort the items and pair heaviest with the lightest.

Can you prove this works?

Table of Contents

General remarks

- Every greedy algorithm has the **greedy choice property**(Reach global optimum from local optimum) and the **optimal substructure property**(Optimal solution to subproblems $=>$ optimal solution to problem)

General remarks

- Every greedy algorithm has the **greedy choice property**(Reach global optimum from local optimum) and the **optimal substructure property**(Optimal solution to subproblems $=>$ optimal solution to problem)
- Hard to prove, easy to code $=>$ just try it (or find a counterexample)