Its Good to Be Greedy

(well...sometimes)

When can we be greedy ?

- Greedy algorithms are generally used in optimization problems
- There is an optimal substructure to the problem
- We do at each step what seems best without planning ahead
- We must be able to prove the correctness of a greedy algorithm if we are to be sure that it works

The Standard Examples

Knapsack Problem - Steal as Much as you can

Interval Scheduling - Complete as many tasks as you can

Aggresive Cows

http://www.spoj.pl/problems/AGGRCOW/

Linear Processing

Sometimes, you can solve a problem in one scan of the input. These are not really greedy problems but can be solved with the same complexity as greedy algorithms.

http://code.google.com/codejam/contest/dashboard?c=1128486

GO !! PRACTICE...

http://www.spoj.pl/problems/AE1B/

http://www.spoj.pl/problems/STAMPS/

http://www.spoj.pl/problems/ARRANGE/

http://www.spoj.pl/problems/BAISED/