Introduction:

The greedy method is perhaps the **most straightforward** design technique, and it can be **applied to a wide variety** of problems.

Most, though not all, of these problems have ***n*** inputs and require us to **obtain a subset** that satisfies some constraints. Any subset that satisfies these **constraints** is called a **feasible solution**. We are required to find a feasible solution that either maximizes or minimizes a given **objective function**. A feasible solution that does this is called an **optimal solution**.

The greedy method suggests that one can devise an algorithm which **works in stages**, considering one input at a time. At each stage, a decision is made regarding whether or not a particular input is in an optimal solution. This is done by considering the inputs in an order determined by some **selection procedure**. If the inclusion of the next input into the partially constructed optimal solution will result in an infeasible solution, then this input is not added to the partial solution.

The selection procedure itself is based on some **optimization measure**. This measure may or may not be the objective function. In fact, several different optimization measures may be plausible for a given problem.

The following procedure describes the greedy method abstractly, but more precisely than above.

```
1   Algorithm Greedy(a, n)
2   // a[1 : n] contains the n inputs.
3   {
4       solution := ∅; // Initialize the solution.
5       for i := 1 to n do
6       {
7           x := Select(a);
8           if Feasible(solution, x) then
9               solution := Union(solution, x);
10      }
11      return solution;
12  }
```

The function SELECT selects an input from A, removes it and assigns its value to x. FEASIBLE is a Boolean-valued function which determines if x can be included into the solution vector. UNION actually combines x with solution and updates the objective function.

Note: Procedure GREEDY describes the essential way that a greedy based algorithm will look, once a particular problem is chosen and the procedures SELECT, FEASIBLE and UNION are properly implemented.

Reference: