

16. Greedy Algorithms

A simple and common way to solve problems and that forms the basis of many algorithms, is the idea of Greed! Sometimes this works really well. At other times it is a really bad thing to do. Consider the survival strategy of the Crocodile. Most animals are fairly picky about their food, only eating their proper food. The Koala bear takes this to the extreme by only eating Eucalyptus leaves. The crocodile does the opposite. It lies waiting in the mud and if anything that moves gets near its mouth – Galumph! it eats it. The idea is that you solve a problem at each decision point by doing the thing that immediately takes you closer to your goal. In the case of the crocodile, rather than wasting time working out whether something really is worthwhile food, just eat it and worry later. How does this apply to other problems? Consider the problem of packing a series of books into a packing crate (perhaps because you are moving). You want to get as many books into the box, but they are all different sizes. Putting the wrong ones in first may lead to a waste of space. A greedy algorithm to do this uses a simple rule: Always put the largest book left in the crate.

While there are books left and the crate is not full **do the following repeatedly**

Put the largest remaining book in the crate.

In what sense is this greedy? Each step uses as much space as possible. It will not necessarily make the best use of the space available but has the advantage of being very quick and simple to follow.

What is the general problem solving rule to follow to use a greedy strategy?

Always take the option that takes you the furthest towards your goal.

One of my favourite sports is Orienteering. This involves finding your way between points marked on a map as quickly as possible. The first time I entered an orienteering contest, I used a greedy algorithm. I always worked out the direction of the next point and tried to run in a straight line towards it – the option that takes me most directly towards the goal of the next point. This is a form of greedy algorithm. Unfortunately it was not necessarily the best algorithm. I spent a lot of time running very quickly. Oddly two girls who seemed to be walking round, seemed to keep getting to the next point before I did. They always seemed to already be there when I arrived (even if I left before they did). They were also keeping to the paths which meant that not only were they going more slowly than me, but they were also planning routes that were further. So how come their algorithm was better than mine? Well even though I was taking the options that most directly took me to my goal, I kept hitting problems along the way – finding a river that I had to wade in my way, or a cliff I had to climb up, not to mention gorse bushes I had to crawl through. So even though I appeared at the point of making the decision to be taking the route that would get me to the destination most quickly, the girls were spending more time planning ahead and pre-processing. They were therefore able to miss all the obstacles I ploughed into blindly. That is the problem of a greedy algorithm, what appears to be taking you most directly to solving the goal may actually contain obstacles or lead you into a blind alley.