

Problems

Dynamic Programming & Greedy Algs

1. On another planet, the denominations of money are:

Denom	1	4	7	13	28	52	91	365
	1	2	3	4	5	6	7	8

optional ~~*~~ a) Show that greed is not always optimal.
(Maybe this is best done with a computer program.)

b) give a recursive formula for the min # of coins to make change = K for any given non-neg K .

You may want to use a helper function.

eg $\text{Opt}(K)$ is function we want

$\text{Opt}(K, i)$ is recursive function we know how
amount \nearrow \uparrow to compute
limit solution
to those that use
 $\text{Denom}[1..i]$

2. Not necessarily contiguous subsequences:

eg DAMN is a subsequence of
DYNAMIC PROGRAMMING

a) $A[1..m]$ $B[1..n]$ arbitrary arrays,

Compute length of longest common subsequence

b) $A[1..m]$ $B[1..n]$

Compute length of shortest common supersequence
of A and B.

c) $X[1..n]$ is **bitonic** if $\exists i$ $1 < i < n$
such that $X[1..i]$ is increasing and
 $X[i..n]$ is decreasing.

Compute longest bitonic subsequence of
an input array A of integers.