

CS375 Small Assignment 8

Due Oct 25th 10PM ET

In class we considered the maximal independent set problem in a path: in this problem, we're given values $v_1, \dots, v_n$ and must choose the subset of these values of maximal combined value such that we don't choose two consecutive indices. First, argue that the following recurrence is accurate: let $\text{OPT}[i]$ denote the optimal combined value from the values $v_i, \dots, v_n$; then $\text{OPT}[i] = \max(v_i + \text{OPT}[i+2], \text{OPT}[i+1])$ for $i < n-1$ and $\text{OPT}[n] = \max(v_n, 0)$ and $\text{OPT}[n-1] = \max(v_{n-1}, \text{OPT}[n])$.

Now that we have the recursive formula, we want to build an array that stores the values. One of the two algorithms works, the other does not. Which is which and why?

```
A = new array of length n + 1
A[n] = max(v_n, 0)
A[n-1] = max(v_{n-1}, A[n])
for i in range(n-2, 0, -1):
    A[i] = max(v_i + A[i+2],
               A[i+1])
```

```
A = new array of length n + 1
A[n] = max(v_n, 0)
A[n-1] = max(v_{n-1}, A[n])
for i in range(1, n-1):
    A[i] = max(v_i + A[i+2],
               A[i+1])
```