

Design and Analysis of Algorithms

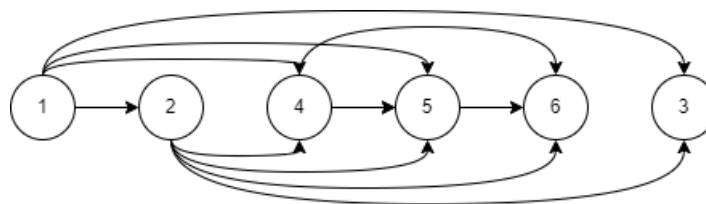
6.3 Dynamic Programming Exercises

Daniel Shannon

May 16th, 2022

6.3.2

- You have a sequence of numbers $x_1, x_2, x_3, \dots, x_n$
- Find the contiguous subsequence $[x_i, \dots, x_j]$ with the greatest sum
- Not allowed to skip elements!
- Use dynamic programming to find an $O(n)$ solution



example linearized DAG

Similar to largest increasing subsequence, but we track the sum instead of the length.

```

prev(0) = x1
for all j = 1, 2, ...n, in linearized order do
    sumj += max{S[j] + xj : (i, j) ∈ E}
    if S[j] < sumj then
        S[j] = sumj
        prev(j) · (i, j) concat
    end if
end for

```