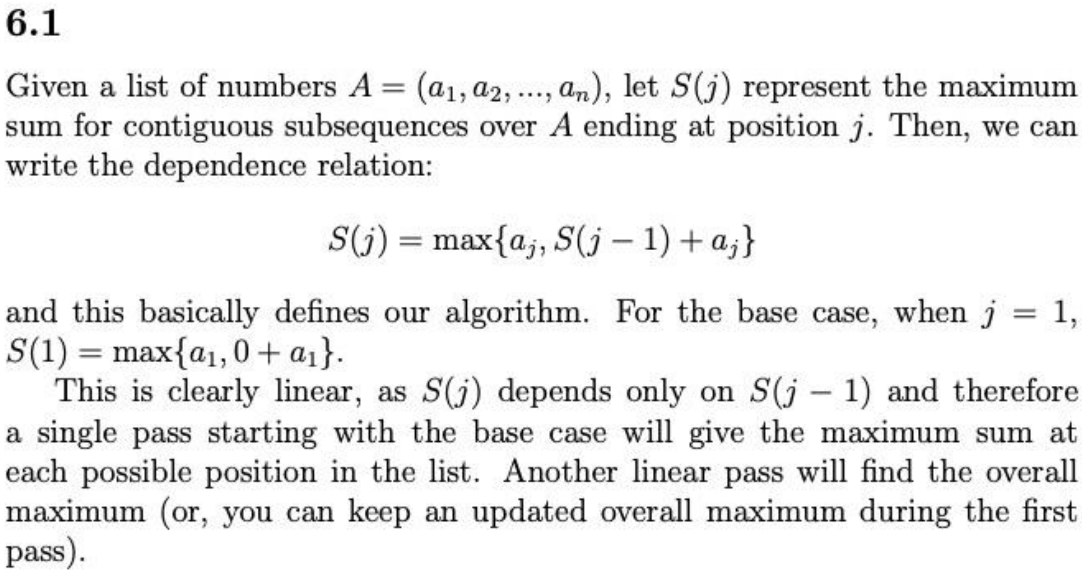
CS 312: Algorithm Analysis

Homework Assignment #14

Show all work neatly.

Question 1: (5) Problem 6.1 from the text



6.1

Given a list of numbers $A = (a_1, a_2, ..., a_n)$, let $S(j)$ represent the maximum sum for contiguous subsequences over A ending at position j. Then, we can write the dependence relation:

$$S(j) = \max\{a_j, S(j-1) + a_j\}$$

and this basically defines our algorithm. For the base case, when $j = 1$, $S(1) = \max\{a_1, 0 + a_1\}$.

This is clearly linear, as $S(j)$ depends only on $S(j-1)$ and therefore a single pass starting with the base case will give the maximum sum at each possible position in the list. Another linear pass will find the overall maximum (or, you can keep an updated overall maximum during the first pass).

Question 2: (5) Knapsack with repetition

Use dynamic programming to fill a knapsack with repetition having a maximum weight capacity of 10 units with a load of maximum value from the following objects:

Object	Weight	Value
A	1	1
B	2	7
C	5	11
D	6	21
E	7	31

Maximum value at weight

w	0	1	2	3	4	5	6	7	8	9	10
K(w)	0	1	7	8	14	15	21	31	32	38	39

The maximum value is 39 and that knapsack configuration comprises {E, B, A}.