

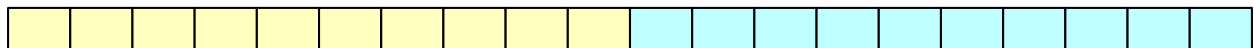
Homework 5

EXTRA CREDIT

Let x_1, x_2, \dots, x_n be a list of positive or negative integers. Give an $O(n)$ divide and conquer algorithm to determine the largest possible sum of a sub-sequence of consecutive items in the list. Implement in Python or Python-like pseudo-code.

Example 1: 10, -20, 3, 4, 5, -1, -1, 12, -3, 1 Largest list has $3+4+5+-1+-1+12 = 22$.

Hints: Obviously, you need to use recursion which needs a base case. When you divide the array into two halves, the best sub-sequence might be



- (a) located entirely in the left half
- (b) located entirely in the right half
- (c) span both sides.

In the latter case, the left side of the longest sequence consists of two parts. The left one is the sub-sequence with largest sum including the end and the right one is the sub-sequence with largest sum including the beginning of the array. So, what you need to do is to calculate with recursion:

- (a) The sub-sequence with largest sum in the array
- (b) The sub-sequence with largest sum in the array starting at the beginning of the array.
- (c) The sub-sequence with largest sum in the array ending at the end of the array.