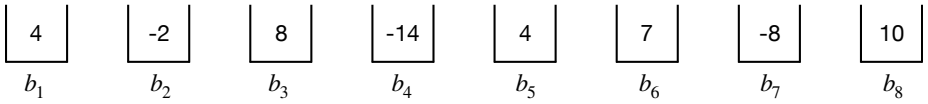
1 Recurrence Use the recursion tree method to solve the following recurrence:

$$T(n) = \begin{cases} T(n/2) + T(n/8) + cn & \text{for } n \geq 2 \\ c & \text{otherwise} \end{cases}$$

2 The Box Game You are a contestant on the game show "Choose Your Boxes Wisely!" You are presented with a line of n boxes $b_1, b_2, \ldots, b_n$, each containing a number $v_1, v_2, \ldots, v_n$. The numbers might be negative. If you choose box b_i you get v_i DKK. You are allowed to choose as many boxes as you want, but they have to be next to each other. Thus, your goal is to choose a sequence of boxes *next to each other* such that the sum S of the numbers in the boxes is maximized. More formally, your goal is to find a contiguous sequence of boxes $b_i, b_{i+1}, \ldots, b_j$ that maximizes $S = \sum_{k=i}^{j} v_k$. We will call such a sequence a best sequence.

In the example below a valid sequence could be b_3, b_4, b_5, which has a sum of -2. The best sequence of boxes below is b_5, b_6, b_7, b_8, which has the sum 13. The sequence b_5, b_6, b_8 is *not* a valid sequence since b_6 and b_8 not are next to each other.



2.1 The game host has precomputed the following information for you: for every number from 1 to n, s_i is the sum of the numbers in box 1 to i. In the above example, we have $s_1 = 4, s_2 = 2, s_3 = 10$, etc.

Let $\textsc{Sum}(i, j) = \sum_{k=i}^{j} v_k$. Explain how you can use $s_1, s_2, \ldots, s_n$ to calculate $\textsc{Sum}(i, j)$ in constant time for any $1 \leq i \leq j \leq n$.

2.2 Describe a divide-and-conquer algorithm that given $v_1, v_2, \ldots, v_n$ and $s_1, s_2, \ldots, s_n$ finds the best contiguous sequence of boxes. Remember to argue that your algorithm is correct.

2.3 Let $T(n)$ be the worst case running time of your algorithm. Give a recurrence for $T(n)$ (and explain why it is correct). What is the asymptotic running time of your algorithm (explain how you obtained the result)?

Hint: A good algorithm runs in $O(n \log n)$ time.