LCRS Algorithms Ch 4 Notes

1 Methods for Solving recurrences

- **substution method,**we guess a bound and then use mathematical induction to prove our guess correct.
- **recursion-tree method** converts the recurrennce into a tree whose nodes represent the costs incurred at vvarious levels of the recursion. We use techniques for bounding summations to solve the recurrence.
- **master method** provides bounds for recurrences of the form

$$T(n) = aT(n/b) + f(n), \tag{1}$$

where $a \geq 1, b > 1, f(n)$ a given function. Note that $f(n)$ is the time of the steps at each level of recursion.

Examples of problems that the master method provides bounds for:

- maximum-subarray problem
- matrix multiplication.

Complexity of Merge Sort algorithm:

$$T(n) = 2T(n/2) + \Theta(n),$$

Because at each step, the algorithm splits the list of numbers in half, and then does n comparisons.

1.1 The Maximum-subarray problem

A brute force solution

Consider a brute force solution to the stock picking problem where we examine every pair of possible buy, sell pairs and then pick the best one. If we consider each possible pair of buy-sell dates, we would consider $\binom{n}{2}$ possibilities, which is $\Theta(n^2)$.

A transformation