15.1-2 We can take the example given in book (Pg. no. 360). For a length of n=4, dynamic programming gives an optimal solution of 2 + 2, which provides us with a revenue of 10. On the other hand, if we chose the piece with maximum density (profit/length), we select the first piece of length 3 (price/length = 2.66), followed by the remaining length of 1. This gives us revenue of 9 (8+1), which is suboptimal.

15.1-3 We can modify the original CUT-ROD algorithm by adding an additional cutting cost. However, this cost is only required when you are actually cutting the rod. Thus when the length of rod is say n, and you make the cut after n, you are not actually cutting the rod, so no cost is required.

modified-BOTTOM-UP-CUT-ROD(p,n)

1. Let r[0..n] be a new array
2. r[0] = 0
3. for j = 1 to n
4.   q = -\infty
5.   for i to j-1
6.     q = max(q,p[i]+r[j-i]-c)
7.   q = max(q,p[j])
8.   r[j]=q
9. return r[n]

15.2-1 The algorithm for matrix-chain multiplication computes the optimal cost: m[1,6]=2010 and s[1,6]=2. It also provides the result that s[3,6]=4. Therefore, the optimal parenthesization is (A_1A_2)((A_3A_4)(A_5A_6)).