1. The sub-problems are: What is the largest value of sum(B), where B is a subsequence of sequence C, a subsequence of A, that ending with the last element of C. C = (a(1), a(2), …, a(i)) for 1 <= i <= n.
2. The base cases are:

opt(1) = a(1)

opt(2) = max{a{1}, a(2)}

The recursion equations are:

opt(i) = opt(i - 2) + a(i), 3<= i <= n.

1. The final answer is: sum(B) = max{opt(i), 1<= i <= n}
2. To obtain n opt(i) need run time O(n). Finding maximum requires run time O(n). Thus time complexity is O(n).