**Assignment 2 Analysis (Alice in ebayland)**

**FIT2014 S2 2018 ID 27799964**

**Task 1: Maximise profit when there is no item limit**

*solve\_task1(product\_list, price\_limit)*

Let N be the total number of products in products.txt, P be the price limit, and S be the size of the solution.

|  |  |  |
| --- | --- | --- |
| **Step** | **Time complexity (worst-case)** | **Space complexity (worst-case)** |
| Create two arrays: one to keep track of the profit at increasing price limits; one to keep track of the solution at each price limit. | O(P) | O(P) |
| For each value of price limit, iterate through each item to find the one that gives the maximum profit, considering previous solutions. | O(PN) | O(1) |
| Output the solution in the format required for the assignment. For each item in the solution, either append it to the output list, or increment its quantity in the output list. Call the function to print the output. | O(S) | O(S) |

Assume S is at most N. Then the overall time complexity is O(PN) and the overall space complexity is O(N+P).

**Task 2: Maximise profit with both item limit and price limit**

*solve\_task2(product\_list, price\_limit, item\_limit)*

Let N be the total number of products in products.txt, P be the price limit, L be the item limit, and S be the size of the solution.

|  |  |  |
| --- | --- | --- |
| **Step** | **Time complexity (worst-case)** | **Space complexity (worst-case)** |
| Create and initialise a (L + 1) x (P + 1) memo array with zeros. Do the same for a solution array of the same dimensions, and initialise all values to None. | O(PL) | O(PL) |
| From memo[1][1], iterate through each column in the first row, filling each cell with the optimal profit for the corresponding item and price limit. Do this for each row.  The optimal profit was calculated by iterating through each item and taking the profit of the item that would give the best overall profit at this cell, plus the value of the cell that corresponds to memo[item limit – 1][price limit – item price].  Record the optimal solutions at each step. | O(PLN) | O(PL) |
| Output the solution in the format required for the assignment. For each item in the solution, either append it to the output list, or increment its quantity in the output list. Call the function that prints the output. | O(S) | O(S) |

Assume S is at most N. If L is considered to be a constant (Alice can only sell 30 items, so L <= 30) then the time complexity is O(PN) and the space complexity is O(PN). If L is not considered to be a constant, O(PLN) time complexity and O(PN) space complexity is evident.