

- Give a URL links to an existing online puzzle/game that is similar to your product:
 - <https://augustineaykara.github.io/Knapsack-Calculator/>
- Indicate if the existing online puzzle/game does provide solutions to users:
 - The existing calculator provides a solution to users, but also requires users to input the problem rather than generating one
- Assume that your program is solving the same problem that are given to users
 - What is your approach (algorithm)?
 - The 0/1 Knapsack Problem can also be solved with dynamic programming, where the solution saves intermediate results while computing the solution. The problem can be broken into subproblems and storing in a 2-D array, then finding the maximum value with the 2-D array.
 - Is your algorithm the optimal ? Why or Why not?
 - This algorithm provides an optimal solution, as it uses dynamic programming.
 - Give the proved or estimated time complexity of your algorithm.
 - $O(N * W)$, where N is the number of items and W is the capacity of the knapsack