Areebah Iqbal
CS 161 Section 01
Professor Tsao
20 February 2024

Week 4: Individual Report

Link to an existing online puzzle/game that is similar to your product:

- https://www.mathsisfun.com/games/math-match-game.html

Indicate if the existing online puzzle/game does provide solutions to users:

- No, the existing online puzzle/game doesn't provide solutions to users.

Approach (Algorithm):

- The user is trying to find two tiles that solve the arithmetic question, so the algorithm will find the two tiles that solve the given problem in the puzzle/game.
- The algorithm that I will use to solve the puzzle/game will start by traversing each value in the grid. While traversing the values, the algorithm will perform an arithmetic operation. Depending on the answer of the arithmetic operation, it will store the answer and values of the array in a hash table. It will continue checking for a pair that solves the arithmetic question until the correct answer is found. Upon discovering the pair of values that correctly solves the arithmetic question, it will flip the corresponding tiles over. Here is the algorithm broken down into steps:
    1. Create a loop to iterate the array and a hash table for storing the iterated values and the solutions.
    2. While iterating the values, perform an arithmetic operation using the target value. For example, if the arithmetic question was addition, the arithmetic operation would be: target - array[i]. Store the solution of the arithmetic operation and the index of the value in the hash table.
    3. Continue iterating the array and performing the operation. If the solution is found in the hash table, save the value and index.
    4. Exit the loop, return the pair, and flip the corresponding tiles over.

Is the algorithm optimal? - Yes, the algorithm is optimal, since the algorithm doesn't revisit the past values in the array. Unlike the brute-force solution, the algorithm visits each value once and only once. It achieves that by utilizing the hash table data structure to store data of visiting values, creating a more efficient and optimal solution.

Time Complexity of the Algorithm: The estimated time complexity is O(n).