Greedy - Task 2

Name.- Santiago Caballero Manzaneda

Formula of the problem (pdf format)

The problem points that there are N accounts, with G values each one, P means the percentage of every account process merge that the bank takes

$2 \le N \le 100000$

$0 \le P \le 20$

$0 \le G \le 10000$

```
for each account in the queue ordered from the lowest to greatest
    it will be merged taking into account the value – percentage value:

  sum = account1.value + account2.value;
  newAccountValue = sum - (sum / 100 * p);
```

Analysis to prove if it is possible to apply Greedy (pdf format)
Greedy Choice

The greedy choice in this problem is to merge the two accounts with the lowest charges at each step. By doing so, we aim to minimize the loss due to the bank's fee. This Is an optimal problem to apply greedy

Optimal substructure

The problem can be constructed efficiently from the optimal solutions to its subproblems. Specifically, at each step of merging two accounts, we make a greedy choice that contributes to the overall optimal solution.

Include Mathematical induction on the prove

1. Base Case: When there are only two accounts n=2, the greedy algorithm chooses to merge these two accounts, minimizing the fee loss. Therefore, the base case holds.

2. Inductive Step: Assuming the correctness of the greedy algorithm for a problem of size \(k \). Proving that this assumption holds for a problem of size \(k + 1 \) by showing that the greedy choice at each step leads to an optimal solution and reduces the problem to a subproblem of size \(k \).

size k.- where k is an arbitrary positive integer, the greedy algorithm produces the correct result.