

Algorithms HW 2

1. Consider the biggest sum problem. The input to this problem is a list of integers and a target integer t . The output is a subset of the list whose sum is as close to t as possible without going over. Prove that the algorithm below does not find the correct elements for every possible input.

Input : values: an array of integers

Input : n : the length of values

Input : t : the target integer

Output : a subset of values whose sum is as close to t as possible without going over

1 Algorithm: GreedySum

2 Sort values in decreasing order (i.e., max-first)

3 sum = 0

4 select = {}

5 for $i = 1$ to n do

6 if values[i] + sum < t then

7 Add values[i] to select

8 sum = sum + values[i]

9 end

10 end

11 return select

Proof:

- We prove that GreedySum is incorrect by counterexample. Consider the following inputs:
 - values = (3,2,1,-1,-2)
 - $n = 5$
 - $t = 10$
- On the first iteration of the for loop $i = 1$, the first element of values “3” and sum “0” are added and compared to the target “10”. Since $3 < 10$ it continues and adds 3 to sum.
- On the second iteration of the for loop $i = 2$, the second element of values “2” and sum “3” added together are $5 < 10$. It again continues since the if condition is fulfilled and the new sum is 5.
- On the third iteration of the for loop $i = 3$, the third element of values “1” and sum “5” are added and compared to target “10”. Since $6 < 10$, it once more enters the if condition and the new sum is 6.
- On the fourth iteration of the for loop $i = 4$, the fourth element of the values is “-1” and sum “6” are summed and compared to target “10”. In this case, $5 < 10$ and it enters the if condition where the new sum is 5.
- On the fifth and final iteration of the for loop $i = 5$, the fifth element of values “-2” is added to sum “5” and compared to target “10”. Like all previous iterations, the if statement is true because $3 < 10$ and the new sum is 3.
- The for loop ends and the result of adding all the sums is 3, which is incorrect because the greatest sum is theoretically adding 3,2,1 to 6, and not including the negative numbers “-1,-2”. Therefore, GreedySort is not a correct sum algorithm.