Middle East Technical University

Department of Computer Engineering

CENG 111

Fall 2015

Take Home Exam 4

v1.1

REGULATIONS

Due date: 23:59, 20 January 2016, Wednesday *(Not subject to postpone)*

Submission: Electronically. You should save your program source code as a text file named `the4.py` and submit it to us via the course's COW page.

Team: There is **no** teaming up. This is an EXAM.

Cheating: Source(s) and Receiver(s) will receive zero and be subject to disciplinary action.

INTRODUCTION

> *"Friendship is everything. Friendship is more than talent. It is more than the government. It is almost the equal of family."* – Vito Corleone

Before his retirement, Don Vito Corleone has requested his bright son, Michael, to take over and run the mafia empire. However, he would not leave any of his savings to his son to run the empire, and ask Michael to take the job only based on his current savings up to when the offer is made. Being puzzled and short in time, Michael *makes an offer you cannot refuse*: He wants your help to check whether his savings are sufficient for taking up the empire.

PROBLEM

> *"What betrayed me? My mind? My heart? Why do I condemn myself so? I swear, on the lives of my children: Give me a chance to redeem myself, and I will sin, no more."* – Michael Corleone

In this THE, your task is to help Michael to make a desperate decision. Michael will give you the mafia empire as a tree, and his savings, and you should determine whether his savings are sufficient for running the mafia empire. In the empire, each person p contributes some amount c_p to the empire and costs w_p to the empire (you may view c_p as the money p collects, and w_p as his wage). Since Michael has a rather more radical nature than his father, if needed, you may cut down the cost w_p of a person by $100 \times r_p$ (i.e., the wage of person p can be forced to $(w_p - 100 \times r_p)$), where r_p is the position of p in the hierarchy of the empire (in tree-terms, r_p is the length of the path from the *root* to p). However, you may reduce the wage w_p of a person only when (i) $r_p > 1$ and (ii) $(c_p - w_p) < 0$.

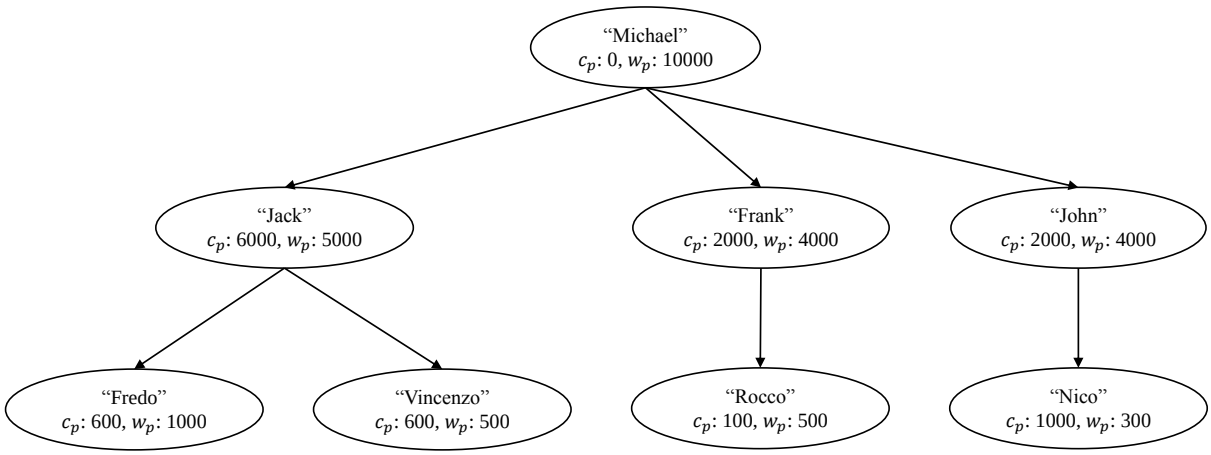

Figure 1: An example mafia tree.

SPECIFICATIONS

- Michael will call your solution via the following function:

  `help_michael(mafia_tree, savings)`

  where `mafia_tree` is the representation of the mafia empire as a nested-list and savings (`int`) is the current savings of Michael.

- The mafia tree will be given to you as a nested list. For the example tree given in Figure 1, the representation will be as follows:

```
[["Michael", 0, 10000], \
   [["Jack", 6000, 5000], [["Fredo", 600, 1000]], [["Vincenzo", 600, 500]]],\
   [["Frank", 2000, 4000], [["Rocco", 100, 500]]],\
   [["John", 2000, 4000], [["Nico", 1000, 300]]]]
```

- As you see in the example, the first item of the mafia tree is the datum. The datum for each node is a list of name (`str`), c_p (`int`), and w_p (`int`). Moreover, note that the tree is not binary.

- Your function should return:

  - `"Yes"` (`str`) if Michael's current savings are sufficient,
  - `"No"` (`str`) if Michael's current savings are insufficient,
  - `["Possible", [p1, w_p1], [p2, w_p2], .., [pN, w_pN]]` if Michael's current savings may be sufficient on the condition that the costs of `p1, ..., pN` are reduced to `w_p1, ..., w_pN` respectively. Note that such reductions can only be possible for a person p if $r_p > 1$ and $(c_p - w_p) < 0$ for p. Note also that there may be several solutions for reduction; you should just offer one of them to Michael.

- Savings s is accepted sufficient if the sum $\sum_{p \in P}(c_p - w_p)$, where P is the list of people in the tree (including Michael), is **less than or equal to** s.
- In case a reduction should be suggested, you should keep both (i) the total reduction amount and (ii) the number of people affected by reduction to a minimum so as to not disturb the peace in the empire.
- It is a must that $c_p > 0$ and $w_p > 0$. Moreover, after reduction, the new wage, w'_p, should satisfy $w'_p > 0$.
- The value $r_p \times 100$ for person p should be interpreted as the upper limit (inclusive). In other words, you may suggest a reduction d_p for a person p such that $d_p \leq r_p \times 100$.
- Each name in the tree will be unique.

EXAMPLE RUN

```
>>> L = [["Michael", 0, 10000], \
   [["Jack", 6000, 5000], [["Fredo", 600, 1000]], [["Vincenzo", 600, 500]]],\
   [["Frank", 2000, 4000], [["Rocco", 100, 500]]],\
   [["John", 2000, 4000], [["Nico", 1000, 300]]]]
>>> help_michael(L, 10000)
"No"
>>> help_michael(L, 12600)
["Possible", ["Fredo", 800], ["Rocco", 300]]
```

NOTES

- You may not use any modules.
- You may not use alternative methods or strategies for representing trees. We, however, encourage you to practice them as an exercise.
- The input, which your functions will be tested with, will be error-free. You do not have to perform any error check.
- Your function(s) should not print anything on the screen.