Code Jam 2014 - Round 3

Analysis: Last Hit

Diana and the tower take turns in shooting N monsters and Diana goes first. Diana can shoot any monster or skip the turn, while the tower always shoots the monster that is the closest to the tower. Each monster i starts with a certain hit points $\mathbf{H_i}$ and it decreases by P when shot by Diana and decreases by Q when shot by the tower. If the hit points goes below 1, the monster dies and cannot be shot further. Diana is awarded $\mathbf{G_i}$ gold if her shot kills the i-th monster, but none if the towerâ€™s shot kills it. What is the maximum amount of gold Diana can obtain?

The key observation here is that, for Diana, shooting a monster other than the one currently targeted by the tower is exactly equivalent to not shooting any monster in this turn and instead getting an â€œextraâ€ shot that can be used at a later turn. Later, Diana may use some or all of her accumulated extra shots consecutively. So instead of making the decision â€œwhich monster do I shoot?â€ before each tower shot, Diana only needs to decide whether to use one of her extra shots (on the closest living monster) or let the tower take a shot.

This reduces the problem to a dynamic programming (DP) solution with the state being: the current monster i to target, the remaining hits points for the current monster and the number extra shots that Diana has. Note that since there is a lower limit on P and Q, the maximum number of extra shots for Diana is **1000**. In the DP solution, there are three transitions:

1. The current monster is dead and we move on to the next monster,
2. Diana skips a shot and gains one extra shot (letting the tower shoot once),
3. Diana shoots the monster once using her extra shots, possibly killing the current monster and getting its gold.

Below is the pseudocode for the top-down dynamic programming with some clarifying comments:

```
# We are at monster i which has rem_hp HP left and Diana has
# extra_shots shots saved up, how much gold can she get?
function rec(i, rem_hp, extra_shots)
  # Base case: all monsters have been killed.
  if (rem_hp <= 0 && i + 1 == N) return 0

  # Monster i is dead, move on to the next one.
  if (rem_hp <= 0) return rec(i + 1, H[i + 1], extra_shots)

  # Memoization.
  if is_set(memo[i][rem_hp][extra_shots])
    return memo[i][rem_hp][extra_shots]

  # The tower shoots next. Diana saves up another shot.
  ret = rec(i, rem_hp - Q, extra_shots + 1)

  # Diana shoots next, using one of the saved up shots.
  # If the shot kills the current monster, she gets its gold.
  if (extra_shots > 0)
    gold = (rem_hp <= P) ? G[i] : 0
    ret = max(ret, gold + rec(i, rem_hp - P, extra_shots - 1))

  return memo[i][rem_hp][extra_shots] = ret
```

```
# Since Diana plays first, she has one extra shot initially.
print rec(0, H[0], 1)
```