Consider the following two-person game.

- There are three piles of rocks (numbered pile 1, pile 2, and pile 3). The starting position is described with a 3-tuple \((i, j, k)\), where there are \(i\) rocks on pile 1, \(j\) rocks on pile 2, and \(k\) rocks on pile 3. Thus, \(i, j, k\) are all non-negative integers and at least one of them is positive.

- The players take turns. In each turn, the player must take at least one rock off, but can only take rocks off of one pile (not two). Furthermore, she can take 1 to \(i\) rocks off of pile \(i\). In other words, she can take one rock off of pile 1, or one or two rocks off of pile 2, or one to three rocks off of pile 3.

- The player who takes the last rock off wins.

**HW 3 (due November 1).** Design a Dynamic Programming (DP) algorithm for this game, whose answer is whether player 1 has a winning strategy. What is the answer for starting condition \((5, 4, 2)\)? (This must be done in latex and the PDF provided in hardcopy.)

**HW 4 (due November 7).** Prove your DP algorithm correct by induction on the total number of rocks in the three piles. (This must be done in latex and the PDF provided in hardcopy.)

**HW 5 (due November 10).** Implement the DP algorithm. Provide commented and well structured code, in any language you like, matching the DP algorithm you designed for HW 3. Show output on 24 starting conditions \((i, j, k)\) where

- \(i \in \{10, 11\}\)
- \(j \in \{20, 21, 22\}\)
- \(k \in \{30, 31, 32, 33\}\)

Your output should be easy to understand, and have sentences that specify the starting condition and whether the first player does or does not have a winning strategy. For example, your output could be:

On starting condition \((1,0,3)\) the first player has a winning strategy.

In the case where there is no winning strategy for player 1, the output could be:

On starting condition \((1,0,1)\) the first player does not have a winning strategy.
or you could have

On starting condition (1,0,1), the second player has a winning strategy.

HW 6 (due November 17). Modify the DP algorithm so that it outputs a winning move for the first player when the first player has a winning strategy. Provide commented and well-structured code, in any language you like. Show output on the same starting conditions as for HW 5. Your output should be easy to understand, and have sentences similar to

On starting condition (1,0,3) the first player has a winning strategy: remove 2 stones from pile 3.

In the case where there is no winning strategy for player 1, the output should be clear about this. For example, you could write:

On starting condition (1,0,1) the first player does not have a winning strategy.

or

On starting condition (1,0,1) the second player has a winning strategy.