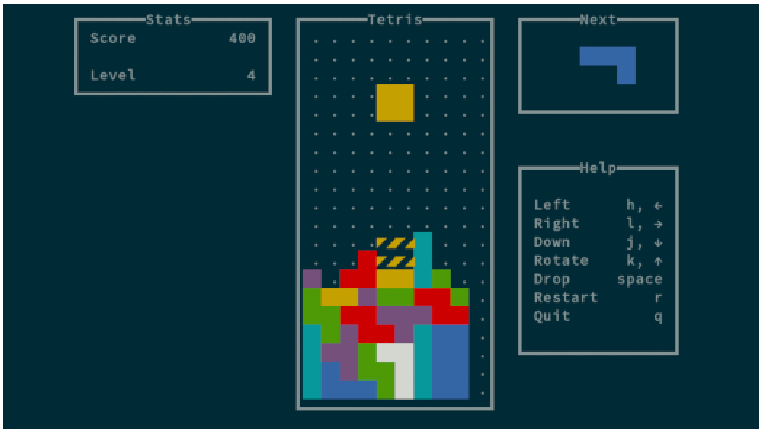
Lab 1

Alexis Wu

Due March 15, 2021 by 12 pm EST

Description

Tetris:



```
Score        400

Level          4
```

```
Left       h, ←
Right      l, →
Down       j, ↓
Rotate     k, ↑
Drop      space
Restart       r
Quit          q
```

Figure 1: Tetris Game

Introduction:

We can rotate (90 degree once) tetris pieces of 7 different shapes to fit into an mxn grids, once a row is filled, that row is cleared.

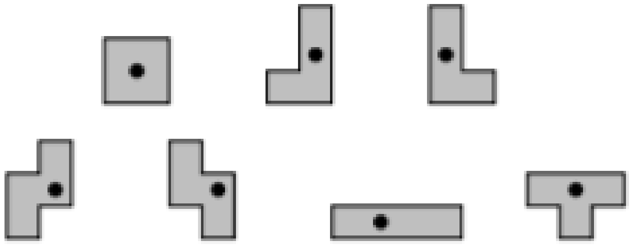


Fig. 1. The tetrominoes Sq ("square"), LG ("left gun"), RG ("right gun"), LS ("left snake"), RS ("right snake"), I, and T, with each piece's center marked.

Figure 2: Tetris pieces of 7 shapes

NP formulation:
We all know the tetris pieces come in a sequence, so we can find out whether we can clear the entire board given a board and a sequence of pieces. This decision takes polynomial time so it is NP-complete.

Proof

Prove Tetris is NP

Suppose we have the game $G = B, p_1, p_2, p_3, p_4, ..., p_{3s}$ where B represents the initial board and p_1 to p_{3s} represents a sequence of tetris pieces. For each piece, it can be in $4 \times B_{size} + 1$ because if it is not landed, it can be at most all grid in B and therefore size B, it can be rotated to 0, 90, 180, 270 degree so there will be 4 rotational states. However, it can also be fixed so each piece in sequence has $4 \times B_{size} + 1$ states and the whole sequence has $3s \times (4 \times B_{size} + 1)$. Checking if this random sequence is feasible takes polynomial time.

0.1 Prove Tetris is NP-complete

To prove that tetris is np-complete we need to perform a reduction from 3-partition:

3 partition problem:

Given a sequence with length of 3s and a target sum T, find if there exists a sequence of combination that the sequence can be regrouped into size of 3 that each group has a sum of T. (each item in sequence value is in range (T/4,T/2)).

To prove tetris is np-complete we must reduce 3 partition problem to tetris by formulating 3 partition into a tetris game.

Initial board

We can translate the 3 partition into if we can clear all rows in a board with residue. The initial board to test satisfiability of 3 partition looks like this:

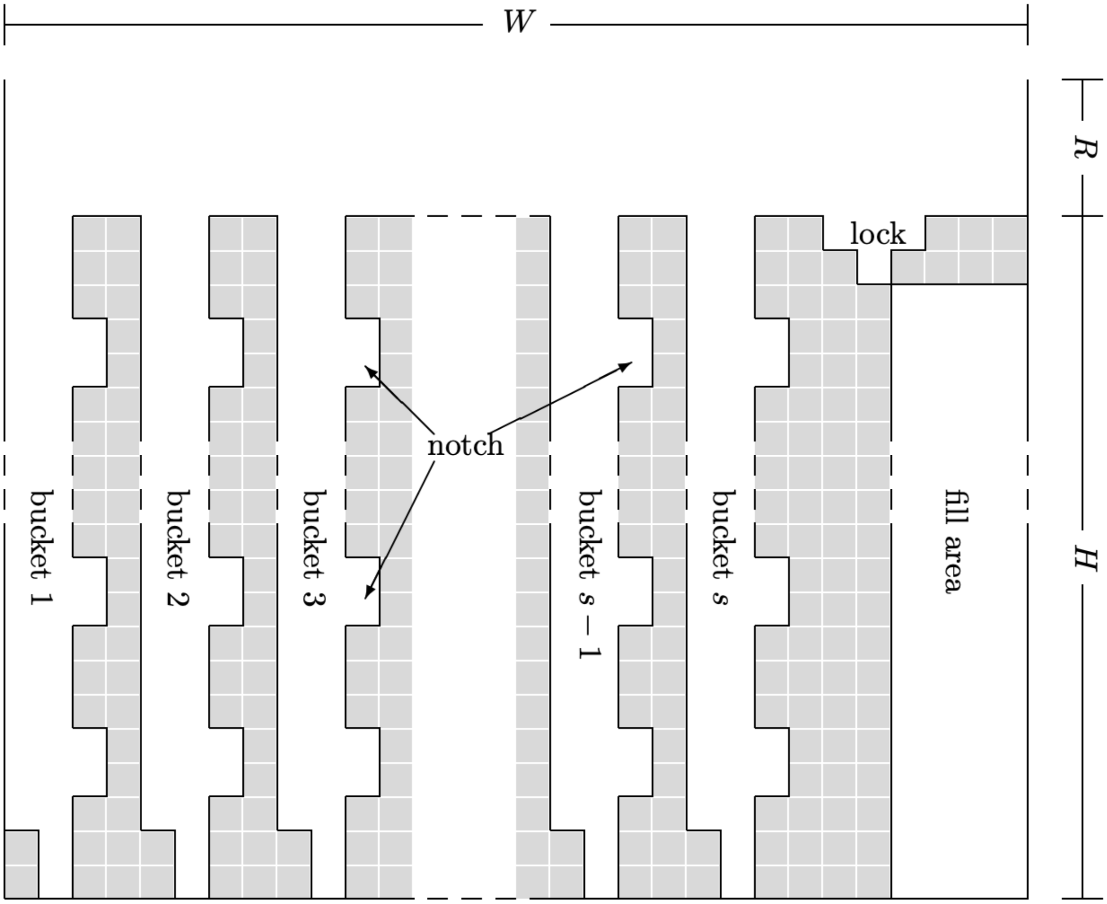


Figure 3: The initial game board.

Specification on initial board (will prove the numbers later):

1. R is the space needed to rotate and translate the pieces. We consider R to be big enough to rotate and translate Tetris pieces above the ‘buckets’ and therefore R is of no consequence to the reduction.
2. W is the width of the game board and is equal to 4s + 6.
3. H is the height of the bottom part of the game board that needs to be cleared and is equal to 5T + 18.

Since the board width and height is linear to the input size, we can construct it in polynomial time.

Fulfilling Questions

Referring back to the graph, we can see there are s buckets each with the column width of 4. On the right, there is a fill area that is locked with the width of 6. This area make sure no rows can be cleared until the top row is filled and the lock is unlocked with the ’T’ shaped piece.

Every bucket represents a subset of size 3 in the 3 partition problem and has T+3 notches on its right.

Sequence of tetris pieces:
For every instance in the 3-partition problem, the corresponding sequence is: an L shaped piece as ’begin’, then followed by p_i times [square,reversed L, square] sequence. Then a square and a line shaped piece as the

’end’.

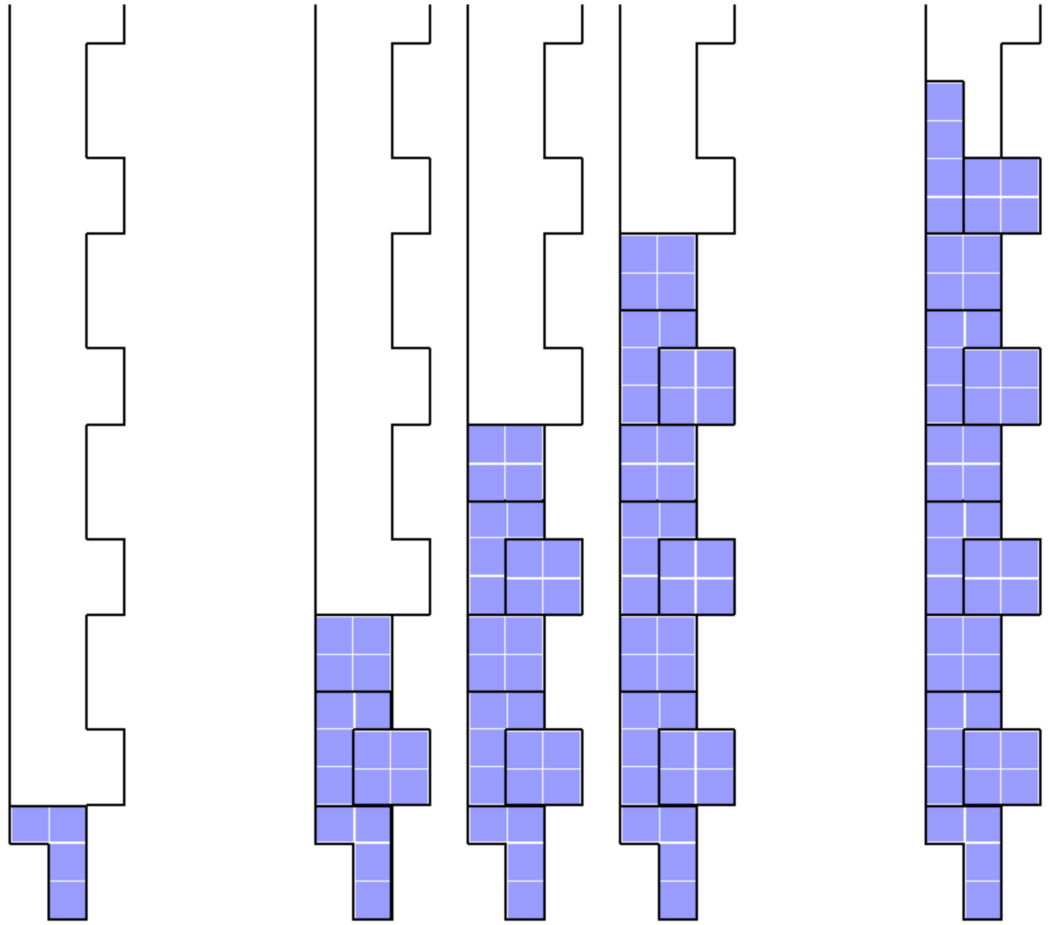


Figure 4: The filling bucket process above.

Then we fill the top of each bucket with L shaped pieces for s times so all the buckets should be filled with the height of 5T+18 because given the process of filling the height should be $5a_i$+5 and max $a_i = TandB_{j_{size}} = 3$

Now we are only left with right most area with lock, this is easy to satisfy and just serve as the purpose to make sure all buckets are filled (all subsets are satisfied) rather than some rows are cleared first. We unlock the lock area with a ’T’ shaped piece and use all the line shaped pieces to fill the right empty columns.

2 Decision Problems Equation

True Condition

When the whole game board is cleared, we assign equal sums to all the buckets (subsets).
subsubsection*False Condition We can test from the bucket that if the pieces are not filled as the described

way above, the buckets cannot be fufilled and cleared.

Implementation

Github

Resources

Tetris is Hard, Made Easy.

Final Note

Tetris is easy to understand (the reduction is not though.) But implementation is not. I will keep working on that.