

Programming Project 11

This assignment is worth 70 points (7.0% of the course grade) and must be **completed and turned in before 11:59 on Wednesday, December 8th, 2010.**

This project will give you experience on design and use of your own classes. We give you less overview on how to do the project, as we want you to design the solution yourself.

Assignment Overview

You are going to design a game system that allows players to play the game of Dominos. As in the solitaire game, you are not playing the game, you are simply supervising that following of rules and scoring. Unlike the solitaire game, you will have to design all the classes and supporting code yourself.

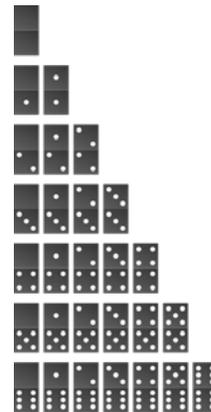
Background

As always, the best thing to do is play the game before you begin writing the solution so that you understand the rules, the goals etc. As it turns out, there are many variations. We are going to play the simplest, the block game. Rules at <http://en.wikipedia.org/wiki/Dominos>, game play at http://www.dominoesonline.com/play_dominoes_online.html.

Classes you must build

There are some similarities you can use to your advantage for this game.

- You are required to build a class Tile, where a tile has two numbers, 0(blank) to six. This is similar to the Card class.
- You are required to build a DoubleSix class, similar to the Deck class. DoubleSix is shown to the right. You can generate this set of tiles, by instantiating the shown tiles with the appropriate values, using the Tile constructor



- You are required to build the LineOfPlay or LOP class. The line of play consists of the sequence of tiles that have played by all the players. The LOP has two ends, the uncovered/exposed numbers of the two end tiles. Plays occur only at those two ends.
- You may or may not want to build a Player class. Not required but recommended

Game Play

Since there are so many variations, we'll play the game shown in the link. 4 players, each with 7 tiles drawn randomly from the DoubleSix deck. First player lays down the first Tile. Each player follows with one of two moves:

- Play a tile from their hand, where one end of their selected tile matches with one of the two ends of the LOP.
- Pass, they cannot play.

Errors

- If they select a tile not in their hand, they should be reprompted for a tile they have
- If the tile they select cannot be played, they should be prompted to select another tile.

Scoring

A player wins if they are the first to play all their tiles. If it occurs that no player can successfully play at the ends of the LOP and all have tiles left, then the game is a draw.

Grading

You are free to design the game (how it looks), but it must be clear and follow the given rules. How well you do this is how well you will be graded.

Deliverables

You must use **handin** to turn in the file `proj11.py` – this is your source code solution; be sure to include your section, the date, the project number and comments describing your code. Please use the specified file name, and save a copy of your `proj11.py` file to your H: drive as a backup.

Notes

There are some challenges here that are not specified:

- How to display a Tile instance and a Players hand as a string. This shouldn't be too hard.
- How to display the LOP. This is a little work as you can wrap the LOP as it gets long. This is not a requirement for the game (just a long sequence is fine) but is something to think about
- The arrangement of a tile in the LOP. A tile can be laid down in one of two directions. If a player picks a tile to lay down, then you must list the tile in its proper orientation: with the same number of the tile against the end of the LOP.