

Lab 4: The Pig Game

You will implement the game of pig, for the user to play against a computer player.

The rules of pig are as follows:

- Two players take turns pushing their luck with a single 6-sided die. Each player may continue so long as they don't roll a 1.
- On a player's turn, the "pool" starts at 0. The player may roll. So long as the roll is not a 1, the value on the die is added to the pool. After each successful roll, the player may choose to keep rolling, or stop and collect the pool.
- If a 1 is rolled, the player's turn immediately ends. The pool is lost and no points are earned.
- The first player with 100 points wins the round.
- There will be two rounds. The human goes first in round 1, and the computer goes first in round 2. (In pig, going first is advantageous—so this helps make things fair.)

Here's how a game might begin:

```
Welcome to Pig!
Round 1!
Human, it is now your turn!
    Your score is 0, and your opponent's is 0.
The pool is now 0.
Do you wish to roll? y
Human rolled a 4.
The pool is now 4.
Do you wish to roll? y
Human rolled a 2.
The pool is now 6.
Do you wish to roll? y
Human rolled a 4.
The pool is now 10.
Do you wish to roll?
```

Start by downloading the files `Pig.java`, `PigPlayer.java`, `HumanPigPlayer.java`, and `ComputerPigPlayer.java`. Your job is to implement three functions within `Pig.java`. You shouldn't modify the others, and you only need to turn in your version of `Pig.java`. (Remember to add your name at the top!)

`Pig.java` contains these three empty functions, that you will need to fill in:

- `static boolean playRound(PigPlayer player1, PigPlayer player2)`, which executes a single round of play. Each player plays a turn in succession, until one of the players earns 100 points. You'll need to keep track of each player's score and the turn

number (0-indexed). When a player wins, print its name and return `true` if it was player 1, and `false` if it was player 2.

- `static int playTurn(PigPlayer player, int turnNum, int score, int opponentsScore)`, which executes a single turn for one of the players. It will continue looping for as long as the player wants to roll the die (which is determined by the player's `decideIfShouldRoll()` method). If the player rolls a 1, it returns 0. If the player stopped rolling before a 1, it should return the points that the player won.
- `static void reportFinalTally(int[] roundsWon, PigPlayer player1, PigPlayer player2)`, which prints to the screen the final score, and congratulates the winning player.

You might find it helps to work first on the `playTurn()` method. You can make `playRound()` so that it only plays a single turn for each player. Then, when you're satisfied that `playTurn()` works, you can change `playRound()` so that it plays many turns.

As always, you must remember to add comments as appropriate. Coding style is important!

Est. 1888

UNIVERSITY *of*
PUGGET
SOUND