PigDice:

|  |  |  |
| --- | --- | --- |
| Behaviour  Our program should handle | Input Example  When it receives | Output Example  It should return |
| 2 players taking turns | hold | Switch player |
| All the rolls in a turn are added to the total when the player stops with a hold | hold | Turn total is added to player total |
| When a player rolls again the roll should be added to their total | Roll again | Their turn total is increased by the roll |
| When a player rolls a 1, they score nothing and it is the other player’s turn | Rolls a 1 | Their turn total is set to zero and switch player |
| Player 2 is done by a computer. | None |  |
| On easy mode it rolls a maximum of two times. On hard it follows a strategy of hold once a score of 20 is reached | Score for the turn <20,  Score for turn =>20 | roll again.  hold |
|  |  |  |

Constructor called Turn

Properties: player1CurrentTurn[array], player2CurrentTurn[array], player1Score, player2Score

Two objects Player1Turn and Player 2 Turn

Every time the Roll button is pressed a random integer from 1 to 6 is added to the xxxxTurn array unless it is a 1, in which case the xxxxTurn array is set to 0 and player switches

If it doesn’t switch because of a 0, it adds the total of the Turnarray to the score.

If player 2 is the computer, on easy mode it rolls twice then holds. On hard mode it rolls until array total => 20 then holds