

Function Test Cases

Function	#	Description	Sample Input Data	Expected Output	Actual Output	P/F
InstructionsPageInput	1	Enables the user to input from the instructions screen (1 to start the same)	1	1 (starts the game)	1 (starts the game)	P
	2	Enables the user to input from the instructions screen (3 to end the game)	3	3 (ends the game)	3 (ends the game)	P
	3	Enables the user to input from the instructions screen. Displays when there is an invalid input and allows the user to input again	2	(displays "invalid input" and user can input again)	(displays "invalid input" and user can input again)	P
InvalidInput	1	Displays when there is an invalid input in the starting screen and enables the user to input again (input 1 to start the game)	1	1 (starts the game)	1 (starts the game)	P
	2	Displays when there is an invalid input in the starting screen and enables the user to input again (input 2 to display instructions)	2	2 (displays instructions)	2 (displays instructions)	P
	3	Displays when there is an invalid input in the starting screen and enables the user to input again.	0	(displays "invalid input" and user can input again)	(displays "invalid input" and user can input again)	P
DifficultyLevel	1	Enables the user to input their desired difficulty level (1 for EASY level)	1	1 (sets level to EASY)	1 (sets level to EASY)	P
	2	Enables the user to input their desired difficulty level (4 for EXTREME level)	4	4 (sets level to EXTREME)	4 (sets level to EXTREME)	P
	3	Enables the user to input their desired difficulty level (Displays when there is an invalid input and allows the user to input again)	7	(displays "invalid input!" and user can input again)	(displays "invalid input!" and user can input again)	P
NumberOfCodePegs	1	Checks and returns the number of code pegs based on your difficulty level (four possible number of code pegs: 6, 8, 10, or 16)	1 (EASY difficulty level)	6 (number of code pegs)	6 (number of code pegs)	P

	2	Checks and returns the number of code pegs based on your difficulty level (four possible number of code pegs: 6, 8, 10, or 16)	4 (EXTREME difficulty level)	16 (number of code pegs)	16 (number of code pegs)	P
	3	Checks and returns the number of code pegs based on your difficulty level (four possible number of code pegs: 6, 8, 10, or 16)	2 (NORMAL difficulty level)	8 (number of code pegs)	8 (number of code pegs)	P
NumberOfAttempts	1	Checks for the number of attempts based on your difficulty level. (four possible number of attempts: 10, 12, 16, or 20)	1 (EASY difficulty level)	10 (number of attempts)	10 (number of attempts)	P
	2	Checks for the number of attempts based on your difficulty level. (four possible number of attempts: 10, 12, 16, or 20)	4 (EXTREME difficulty level)	20 (number of attempts)	20 (number of attempts)	P
	3	Checks for the number of attempts based on your difficulty level. (four possible number of attempts: 10, 12, 16, or 20)	2 (NORMAL difficulty level)	12 (number of attempts)	12 (number of attempts)	P
NumberOfGames	1	Scans and checks the validity of the number of games (the number must be positive, even, and not greater than 10)	2	2 (number of games)	2 (number of games)	P
	2	Scans and checks the validity of the number of games (the number must be positive, even, and not greater than 10)	6	6 (number of games)	6 (number of games)	P
	3	Scans and checks the validity of the number of games (Displays when there is an invalid input and allows the user to input again)	5	(displays “invalid input!” and user can input again	(displays “invalid input!” and user can input again	P
AssignRole	1	Scans and checks the validity of the role the player starts as (‘B’ for codebreaker)	‘B’	‘B’ (player’s role)	‘B’ (player’s role)	P
	2	Scans and checks the validity of the role the player starts as (‘M’ for codemaker)	‘M’	‘M’ (player’s role)	‘M’ (player’s role)	P
	3	Scans and checks the validity of the role the player starts as (Displays when there is an invalid input and allows the user to input again)	‘b’	(displays “invalid input” and user can input again	(displays “invalid input” and user can input again	P

MakeCode	1	Generates a character for each code peg using rand and upper limit	6 (number of code pegs)	'R', 'P', 'G', 'Y', 'O', 'S' (only one of the characters would be generated)	'R', 'P', 'G', 'Y', 'O', 'S' (only one of the characters would be generated)	P
	2	Generates a character for each code peg using rand and upper limit	16	'R', 'P', 'G', 'Y', 'O', 'S', 'A', 'N', 'T', 'E', 'V', 'M', 'C', 'I', 'F', 'J' (only one of the characters would be generated)	'R', 'P', 'G', 'Y', 'O', 'S', 'A', 'N', 'T', 'E', 'V', 'M', 'C', 'I', 'F', 'J' (only one of the characters would be generated)	P
	3	Generates a character for each code peg using rand and upper limit	8	'R', 'P', 'G', 'Y', 'O', 'S', 'A', 'N' (only one of the characters would be generated)	'R', 'P', 'G', 'Y', 'O', 'S', 'A', 'N' (only one of the characters would be generated)	P
ColorOptions	1	Displays the options for the code pegs based on the difficulty level and role	1, 'B' (the integer represents the difficulty level and the character represents the player's role)	"You are the codebreaker\n(Use R P G Y O S for the pegs)" (message is printed)	You are the codebreaker\n(Use R P G Y O S for the pegs)" (message is printed)	P
	2	Displays the options for the code pegs based on the difficulty level and role	1, 'M'	"(Use R P G Y O S for the pegs with spaces in between):\t" (message is printed)	"(Use R P G Y O S for the pegs with spaces in between):\t" (message is printed)	P
	3	Displays the options for the code pegs based on the difficulty level and role	4, 'B'	"You are the codebreaker\n(Use R P G Y O S A N T E V M C I F J for the pegs)" (message is printed)	"You are the codebreaker\n(Use R P G Y O S A N T E V M C I F J for the pegs)" (message is printed)	P
ValidInput	1	Checks the validity of the inputs of the Code Guesser and Maker based on the difficulty level. (returns 1 if valid and 0 if invalid)	1, 'G' (the integer represents the difficulty level and the character is the input for the code peg)	1	1	P
	2	Checks the validity of the inputs of the Code Guesser and Maker based on the difficulty level. (returns 1 if valid and 0 if invalid)	2, 'A'	1	1	P
	3	Checks the validity of the inputs of the Code Guesser and Maker based on the difficulty level. (returns 1 if valid and 0 if invalid)	1, 'E'	0	0	P
CorrectPegs	1	Counts the number of correct guesses	'G', 'Y', 'O', 'S', 'O', 'Y', 'R', 'S' (the first 4 characters represent the code pegs and the last 4 are code guesser's guesses for the code pegs)	2	2	P

	2	Counts the number of correct guesses	'R', 'P', 'Y', 'G', 'R', 'P', 'Y', 'G' (the first 4 characters represent the code pegs and the last 4 are code guesser's guesses for the code pegs)	4	4	P
	3	Counts the number of correct guesses	O', 'Y', 'R', 'S', 'R', 'P', 'Y', 'G' (the first 4 characters represent the code pegs and the last 4 are code guesser's guesses for the code pegs)	0	0	P
CheckForW	1	Checks for misplaced pegs and displays W for each	'G', 'Y', 'O', 'S', 'O', 'Y', 'R', 'G' (the first 4 characters represent the code pegs and the last 4 are code guesser's guesses for the code pegs)	" W W" (prints W for every misplaced peg)	" W W" (prints W for every misplaced peg)	P
	2	Checks for misplaced pegs and displays W for each	'R', 'P', 'Y', 'G', 'R', 'P', 'Y', 'G' (the first 4 characters represent the code pegs and the last 4 are code guesser's guesses for the code pegs)	(nothing is printed as there are no misplaced pegs)	(nothing is printed as there are no misplaced pegs)	P
	3	Checks for misplaced pegs and displays W for each	O', 'Y', 'R', 'S', 'R', 'S', 'Y', 'O' (the first 4 characters represent the code pegs and the last 4 are code guesser's guesses for the code pegs)	" W W W W" (prints W for every misplaced peg)	" W W W W" (prints W for every misplaced peg)	P
DisplayResult	1	Displays the overall result of the game	21, 14 (the first integer represents the human player's score while the second integer represents the AI player's score)	"Congratulations! You won!" (prints this message if the human player's score is greater than the AI player's score)	"Congratulations! You won!" (prints this message if the human player's score is greater than the AI player's score)	P
	2	Displays the overall result of the game	16, 28	"You lost!" (prints this message if the human player's score is less than the AI player's score)	"You lost!" (prints this message if the human player's score is less than the AI player's score)	P
	3	Displays the overall result of the game	20, 20	"It's a draw!" (prints this message if the human player's score is equal to the AI player's score)	"It's a draw!" (prints this message if the human player's score is equal to the AI player's score)	P

playAgain	1	Displays options and allows the user to either play again, display instructions, or exit the game (1 to start the same)	1	1 (starts the game from the beginning)	1 (starts the game from the beginning)	P
	2	Displays options and allows the user to either play again, display instructions, or exit the game (3 to end the game)	3	3 (ends the game)	3 (ends the game)	P
	3	Displays options and allows the user to either play again, display instructions, or exit the game (Displays when there is an invalid input and allows the user to input again)	7	(displays “invalid input” and user can input again)	(displays “invalid input” and user can input again)	P