

Branching Assignment (Dice Game)

Due date: _____

Objectives

The purpose of this assignment is to assess your understanding of the following concepts:

- Branching (if-else)
- Boolean expressions
- Comparison operators (<, >, ==, <=, >=, !=)
- Logical operators (&&, ||, !)

Remember, if you're spending a lot of time on something that isn't part of the assignment objectives, then get some help. Search the internet first, then ask a neighbor. If you still can't figure it out, then ask Mr. Brooks.

Assignment

For this assignment, you will write a dice game.

The game window has four buttons representing the four dice in the game. The game is played in ten rounds, with the total score accumulating with each round. The final score is the total score at the end of the tenth round.

The game window also shows which round we're on, the number of rolls left for this round, the current round score, and the total (banked) score for the game.

The "round score" is the sum of all of the dice. If all four dice are the same number, then the round score should be five times the sum. (For example, rolling all fives gives a round score of 100.)

At the beginning of each round, the game rolls all four dice. The user can then choose to re-roll individual dice by clicking the corresponding button, or to bank the round score and start the next round. (Look at the guess-the-number game we wrote in class to see how to generate a random number.)

The user can only re-roll up to 5 times per round. After 5 re-rolls, clicking the dice buttons does nothing.

When the user clicks the "New Round" button to start the next round, the game should add the current round score (sum of all the dice) to the total game score, then roll all four dice and start the next round (the number of re-rolls goes back to 5).

When the user clicks the "New Round" button to end the tenth round, the game should update the total score as usual, then change the round number or round score text to say "Game Over". The game should also disable all of the buttons (set the `IsEnabled` property to false).



Initial focus: The start of the game rolls all four dice and calculates the round score. Clicking on a dice button rolls that dice and re-calculates the round score. After five rolls clicking on a dice button does nothing.

Points Breakdown

- 10 points: The game window has all of the necessary controls and meaningful variable names (including control names) that use the “camelCase” naming convention.
- 10 points: The user can re-roll any die by clicking on the corresponding button.
- 20 points: The game only allows five re-rolls per round (no matter which dice are rerolled). After that, clicking a dice button does nothing. (You should not achieve this by disabling the dice buttons.)
- 20 points: The game updates the round score whenever any dice are rolled (or re-rolled). The round score is the sum of all the dice, unless all four dice have the same number, in which case the round score is five times the sum.
- 20 points: When the user clicks the “New Round” button, the game adds the current round score to the total score and starts the next round (increments the round number and rolls all four dice).
- 20 points: At the end of the tenth round (when the user clicks the “New Round” button), the game shows the new total score then ends the game by displaying “Game Over” and disabling all of the buttons.

Additional Learning

- (1 ticket) Change the dice buttons to use images instead of text to indicate the dice value. (Use an **Image** control inside the button to do this. I will provide more information in a bonus class about adding images to your programs.)
- (1 ticket) Add a button that starts a new game (resets round and scores to zero) after the game is over.
- (1 ticket) After implementing a button to start a new game, add a list of 5 high scores. At the end of the game, if the total score is high enough to make the list, the list is adjusted to include the new score.