

CSC 241

Lab 1

Complete the **BowlingGame** created in **Exercise 1**.

Now that your **shot** method is working, you can implement the **computeScoreFrame** method. This method will tally up the score of the game after each frame.

In order to keep the score of the game updated frame-by-frame, you will have to consider the three following special cases as you do so:

Case 1 (1 spare before the current frame):

throw 1	throw 2	score	computation for amount to add to score
2	8	10	$2+8=8$
6	3	25	$6*2+3=15$

Case 2 (1 strike before the current frame):

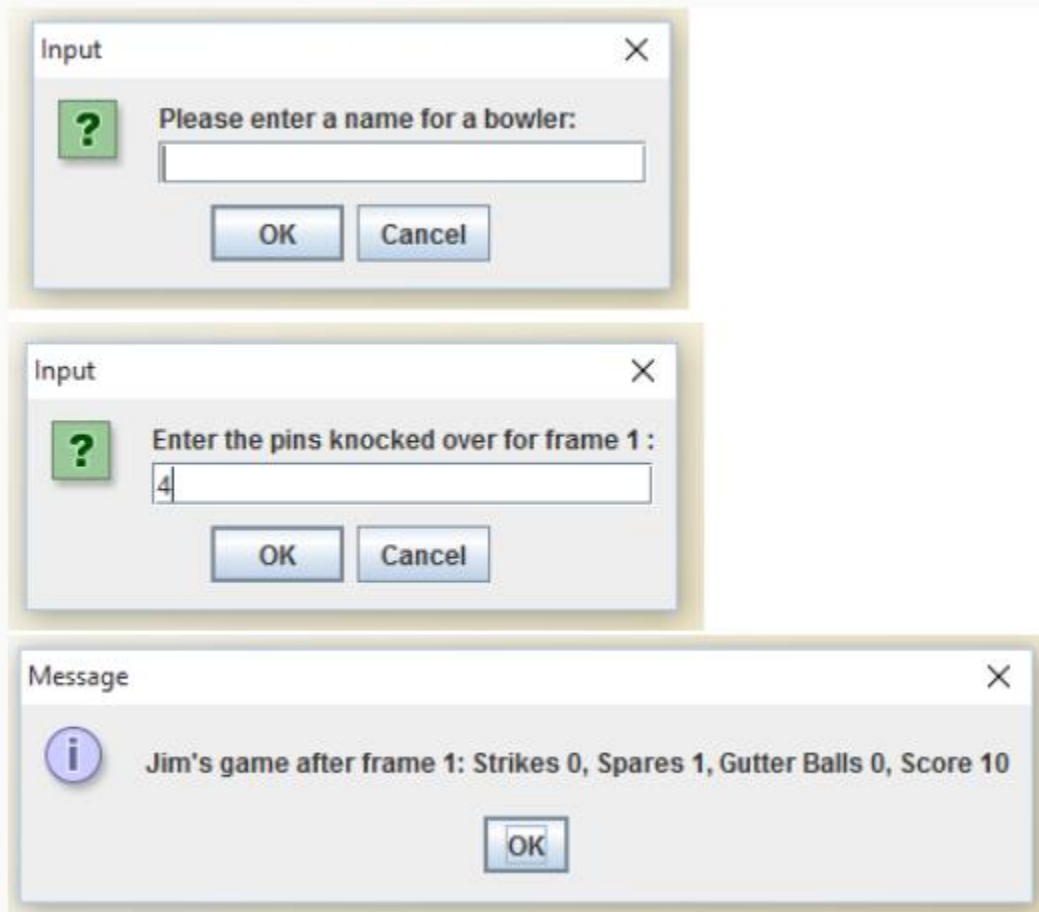
throw 1	throw 2	score	computation for amount to add to score
X	-	10	$X=10$
5	4	28	$(5+4)*2=18$

Case 3 (2 strikes before the current frame):

throw 1	throw 2	score	computation for amount to add to score
X	-	10	$X=10$
X	-	30	$X+X=20$
3	4	47	$(3*2+4)+(3+4)=17$

Notice that, the current score after the frame is bowled requires knowledge of the previous frames' scores. That is why we maintain the scores in the shot method. That is why these three cases depend on the previous frames.

Here are some screen shots of the game being played:



Try the following game and verify the total score is: 158

FRAME:	1	2	3	4	5	6	7	8	9	10
SHOTS:	2 6	X X	5 /	2 -	7 1	X X	5 /	X 3 5		

Extra Credit +5 points:

Write a method called **computeScore** that finds the score of a bowling game after all of the frames have been bowled. Test the method after the game is bowled and verify the score is the same as the final display given by `computeScoreFrame`, after the 10th frame is bowled.