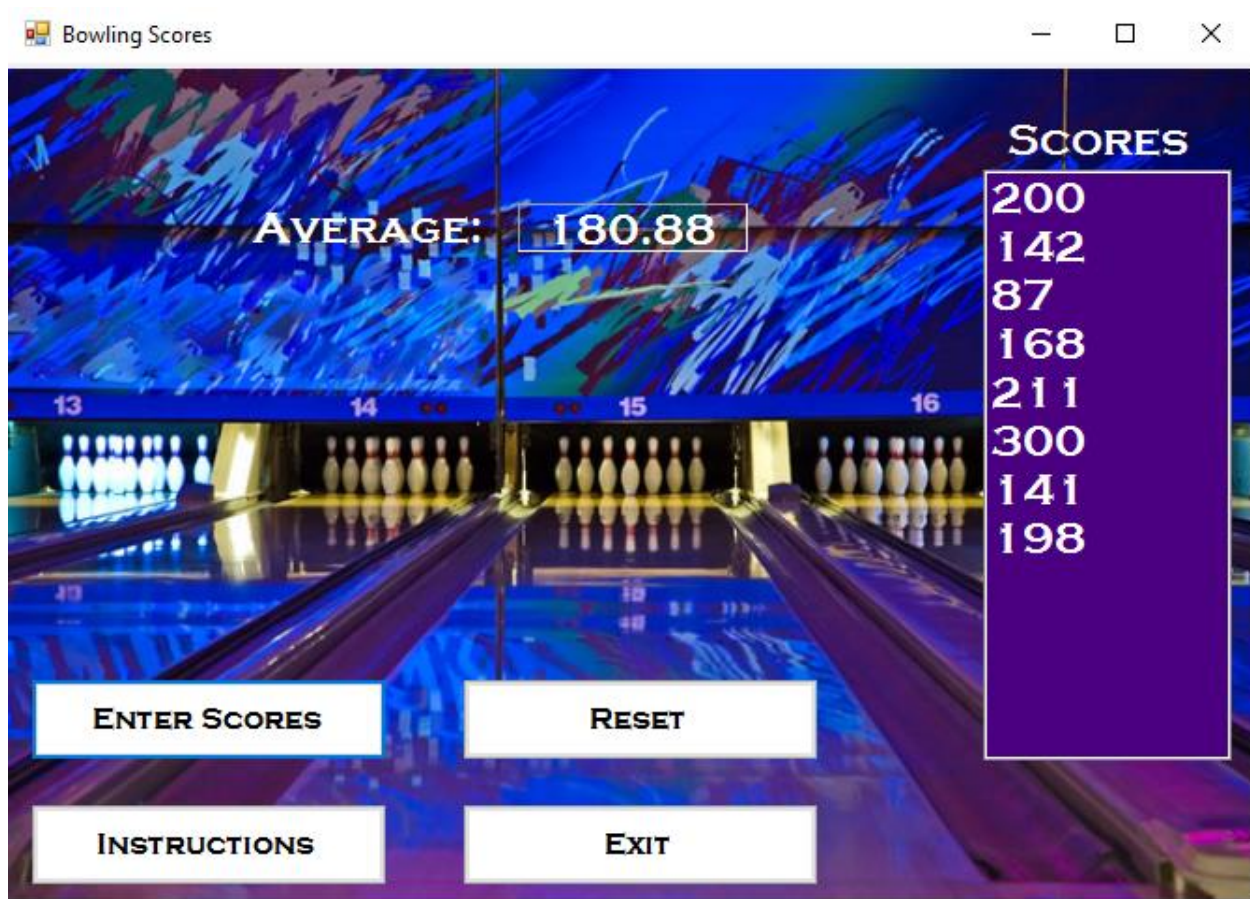


Lab 5A – Bowling Scores

You will create a program that allows users to enter bowling scores. Once the scores are entered, they appear in a ListBox and the average of the scores displays on the form. Users can then double-click on a score to see if it was a perfect game (score of 300).



Objectives

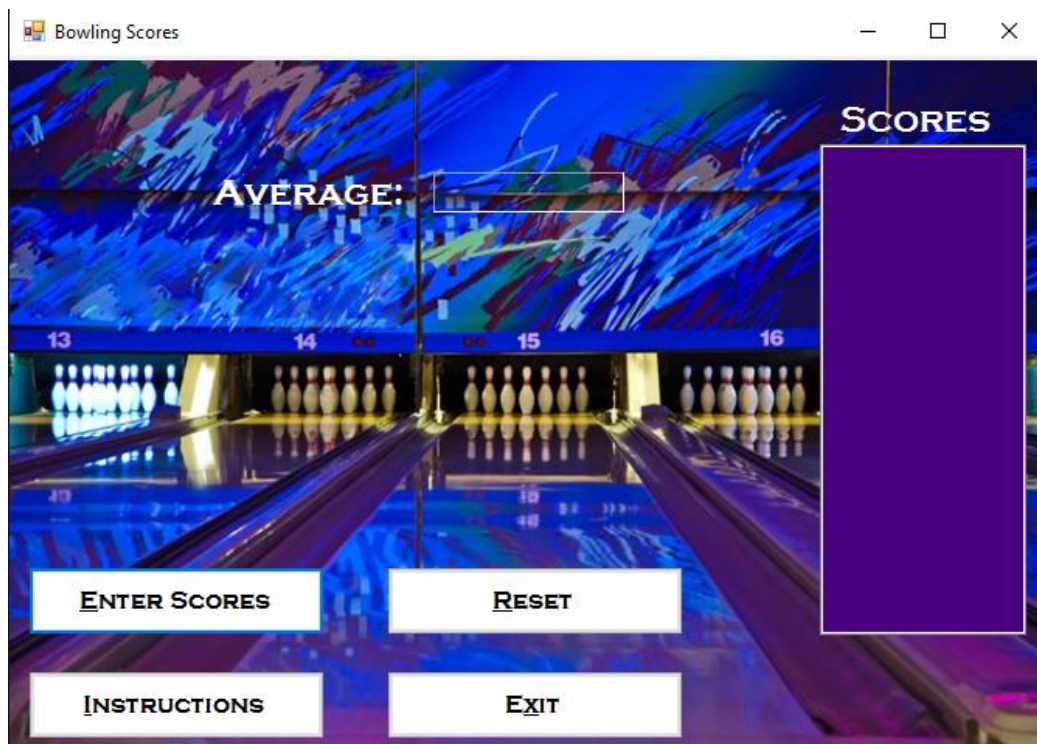
- Implement a `Do... Loop`
- Use of a `ListBox`
- Use of `ToolTip`
- Use of accumulators (running totals) and counters
- Obtaining data from an `InputBox`

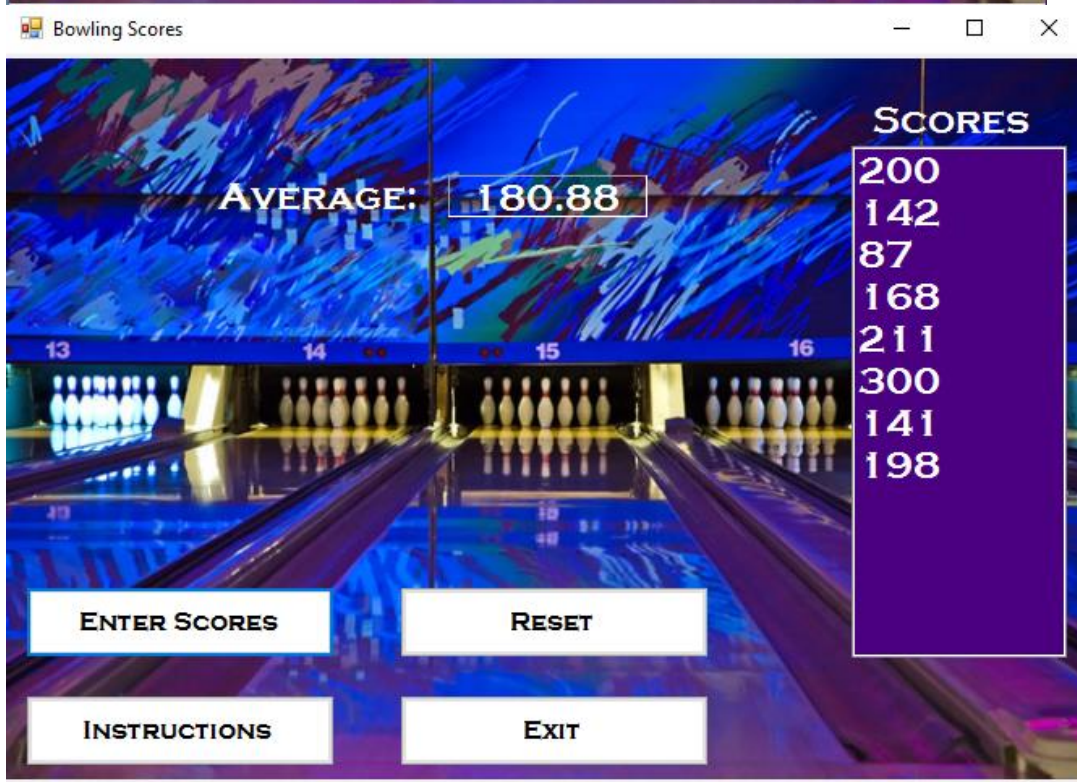
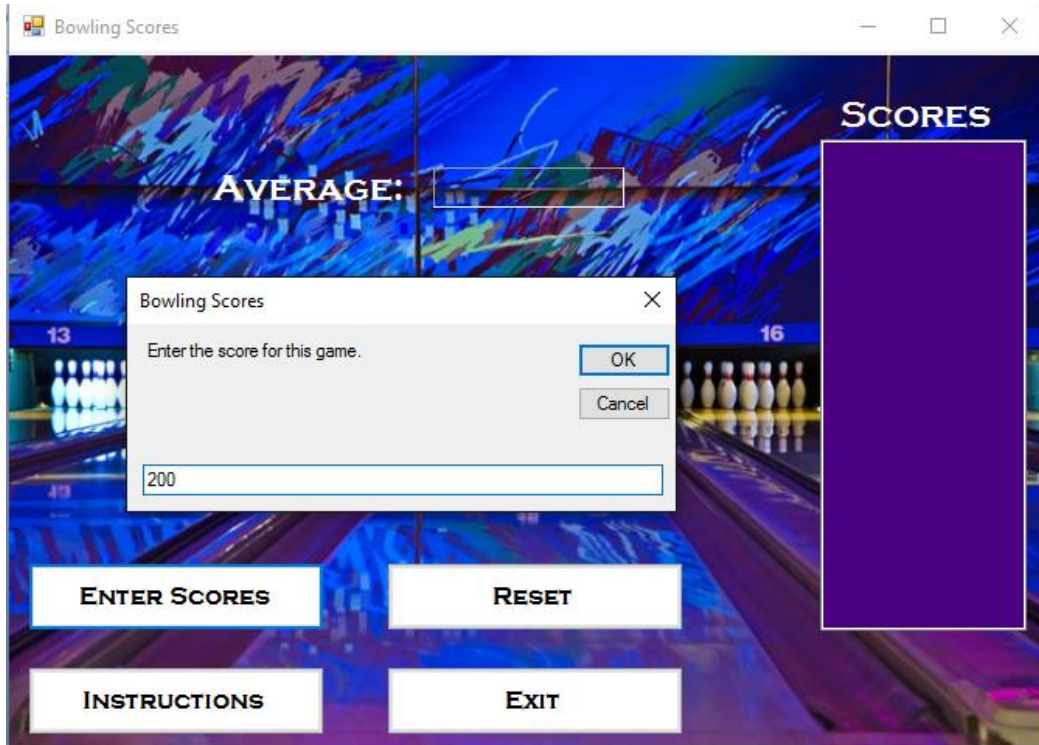
Requirements & Tips

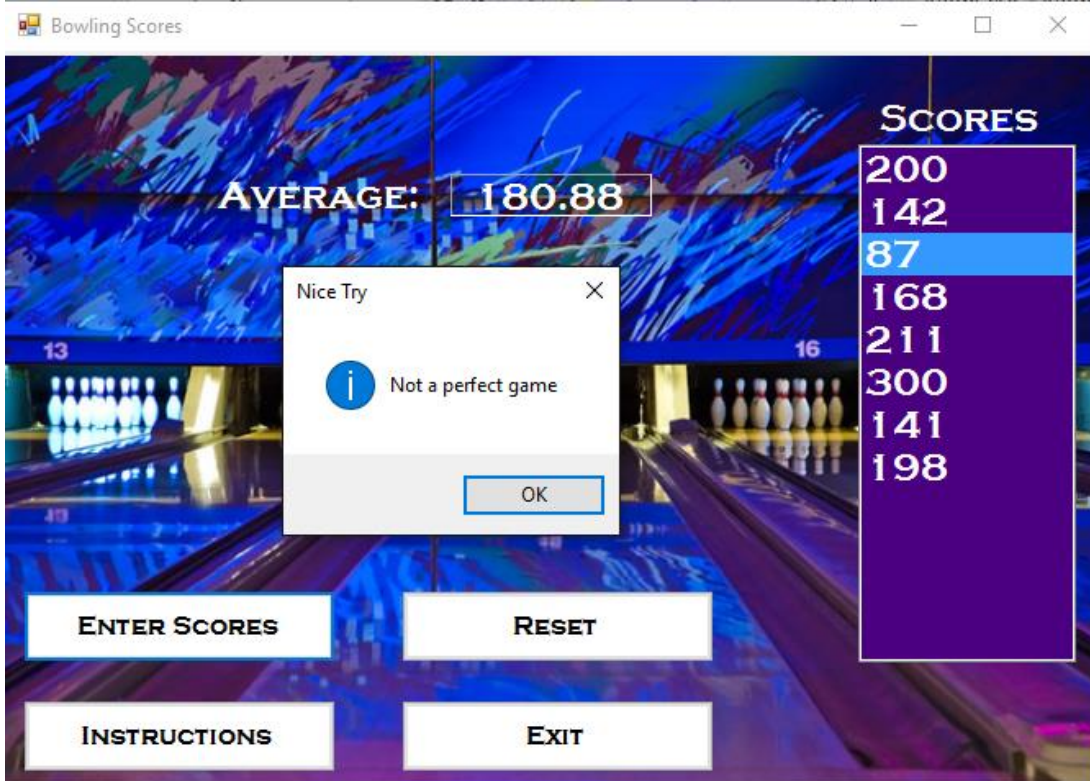
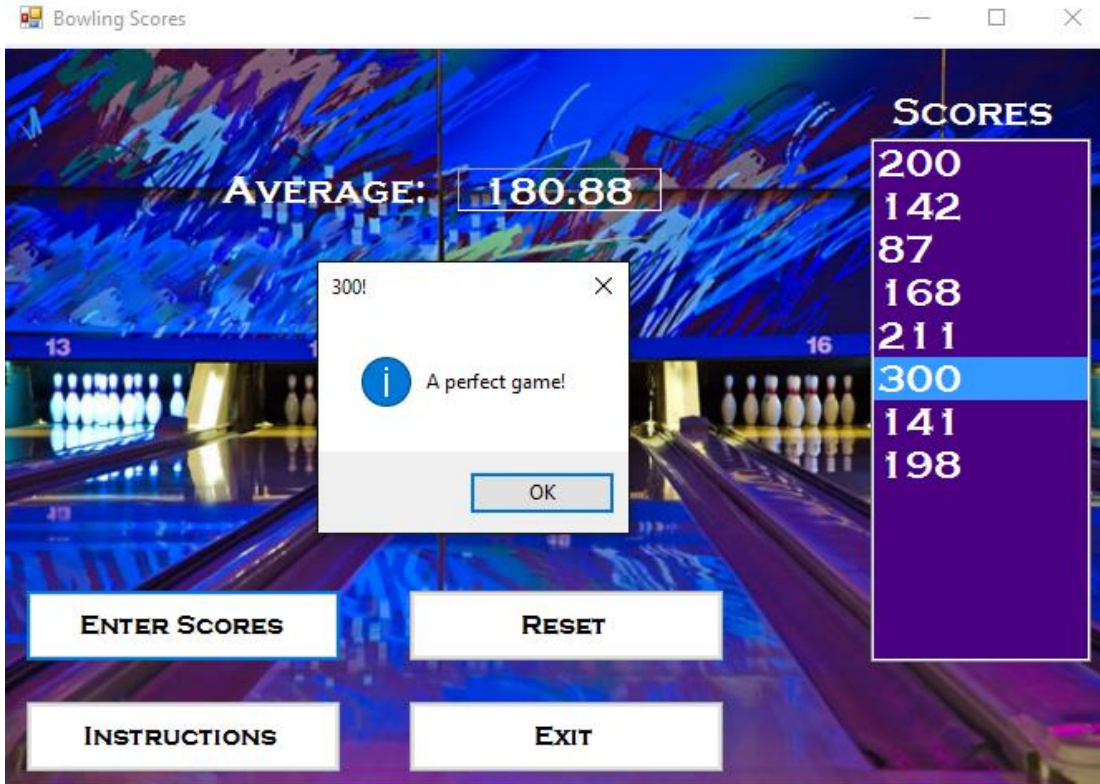
- Comment your code!!!
- You will want two global variables that keep track of the total games played and the total pins knocked down.
- Display the instructions in a **MessageBox** when players click an INSTRUCTIONS button.
- End the program with an EXIT button.
- When users click the ENTER SCORES button...
 - Use a **Do... Loop** to continually bring up an **InputBox** allowing users to enter a bowling score. This loop will continue until users click CANCEL (that makes the **InputBox** return a value of **String.Empty**)
 - In the loop, make sure the value returned by the **InputBox** isn't **String.Empty**. If it isn't, then users didn't press CANCEL and want to enter a new bowling score.
 - Make sure the value entered in the **InputBox** can be converted to an integer. If it can't, provide an error message for the user.
 - Make sure the integer value is a legal bowling score (the lowest bowling score is 0; the highest bowling score is 300). If it isn't, provide an error message for the user.
 - Once you have a valid score...
 - Put the new score in the **ListBox**
 - Update the total pins and the total games.
 - Calculate and display the average (you can do this either in or out of the loop). The average should be formatted to display to two decimal places.
- When users click the RESET button...
 - The scores in the **ListBox** and the average score should disappear.
 - The variables holding total pins and total games must be reset
- Something new! You will code a double-click event for the **ListBox** – you can view a video on this topic at
 - Screencast (<http://screencast-o-matic.com/u/etC7/LBoxDbIClk>)
 - YouTube (<https://youtu.be/13uE25745uA>),
 - Or download an MP4 file (<http://bowerpower.net/compprog1/vb05/LBoxDbIClk.mp4>).
 - The shell program for the videos is at <http://bowerpower.net/compprog1/vb05/LBoxDbIClk.zip>.
 - Make sure the number of items in the **ListBox** is more than zero. If it is...
 - When users double-click on a score of 300, a message appears that this is a perfect game.
 - When users double-click on a score that isn't 300, a message appears indicating this is not a perfect game.
- Tab order: ENTER SCORES button, the rest of your buttons (you select a logical order), the **ListBox** (make the **ListBox** the last control in the tab order)
- Include keyboard access to the four buttons and tool tips for the buttons and **ListBox**.
- **AcceptButton** is ENTER SCORES; **CancelButton** is optional

- Type and modify the following at the top of the code:
'Your Name
'Date
'Assignment Name (Lab 5A - Bowling Scores, for example)
'Computer Programming I - Bower (or CS 11400 - Bower if you want IPFW class)
'This program will... (describe the program)

Sample Data (remember that the access key underscores only appear when users press ALT – they are only displayed in the first image)







Grading

| | |
|--|-----------|
| Form design/ Naming standards | 4 |
| Runs properly | 3 |
| Comments/documentation included | 3 |
| Enter scores button | 3 |
| Instructions/Help button | 1 |
| Reset/Clear button | 2 |
| Exit/End button | 1 |
| Check for valid input | 2 |
| Do... Loop | 2 |
| ListBox double-click event | 1 |
| Accurate results: average | 3 |
| Accurate results: perfect game | 2 |
| ToolTip | 1 |
| Run-time communication with user (includes use of InputBox) | 2 |
| TOTAL | 30 |