Information Systems 107
Programming and Application Development

Programming Project 9

Project Directions

This is an individual programming project.

For this project you will create a multi-form application called Bowling Match to enter scores for a bowling match between four teams with four bowlers per team each bowling three games. One form will be used to enter a team name and the 12 game scores for that team. When the data for one team is entered on the form, the Record Scores button will be clicked to save the data in a three-dimensional array. After entering the first team's scores and saving them, the form will be cleared so that the scores for the second team can be entered, and so forth, until all four team scores have been saved. After the fourth team's scores are entered, the form will close and return to the main form that is used to display the match results. Clicking the Display Match Results button will process the score data and identify the winning team name, winning team total score, and the highest individual game score.

Choose meaningful names for the forms, form controls, global public variables, and other procedure variables. Example forms and programming suggestions are included on the next pages.

Grading

Hard copy

- You do not need to print out any part of your project for grading.

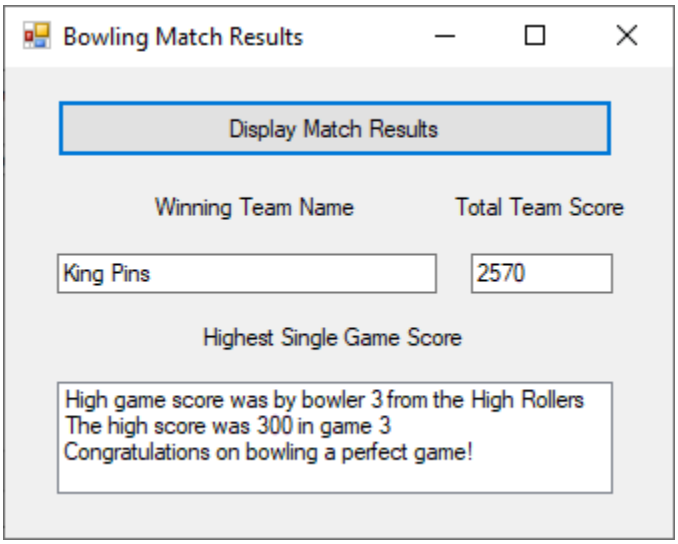
Executable files

- Create a folder named LastNamePP9 (in zip format) that will contain all of the files for your project. Your last name is included in the folder name so that each folder name is unique.
- Upload your zipped folder in the course Blackboard page content area in the assignment named Upload Completed PP9 Folder.

110 points possible

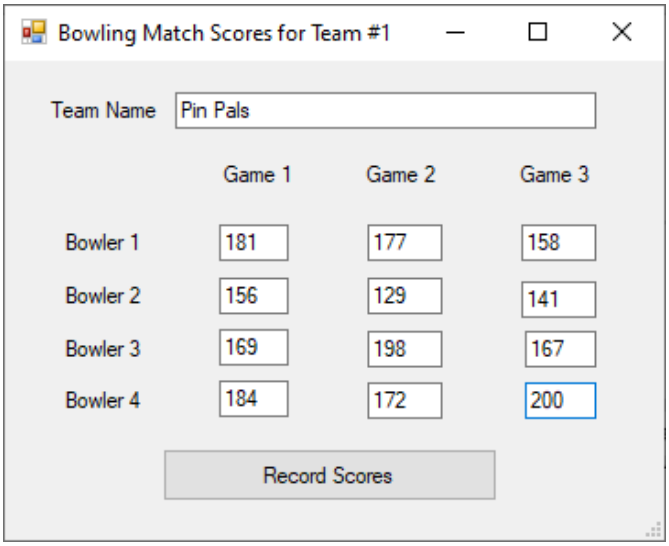
Primary form used to display the bowling match results

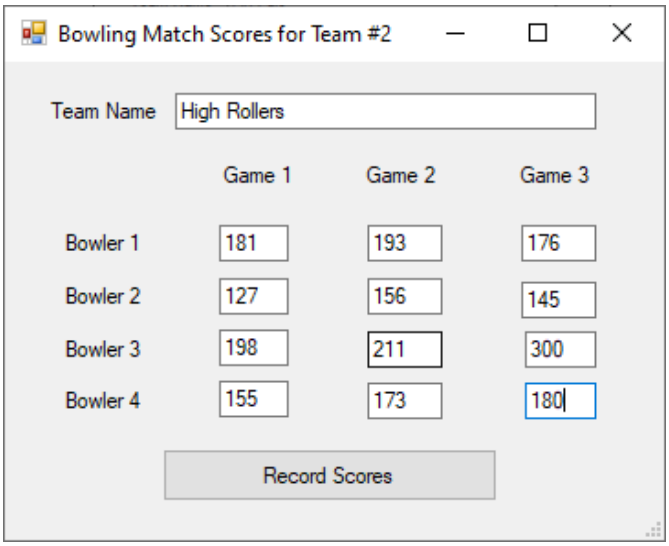
- Includes one button, three labels, two text boxes, and one list box.

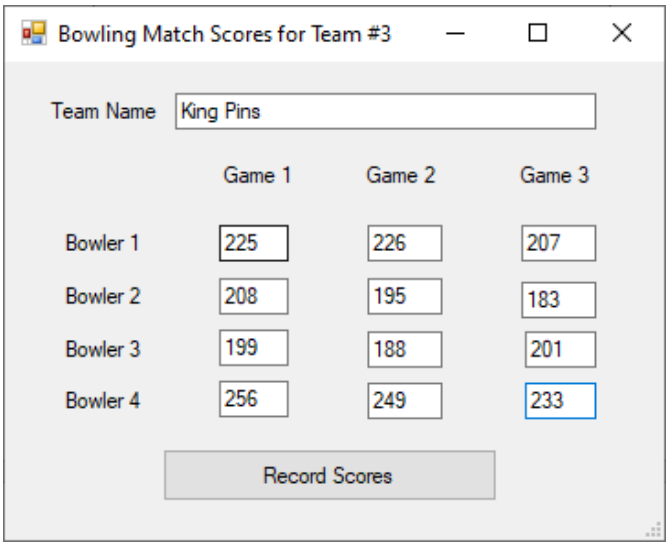


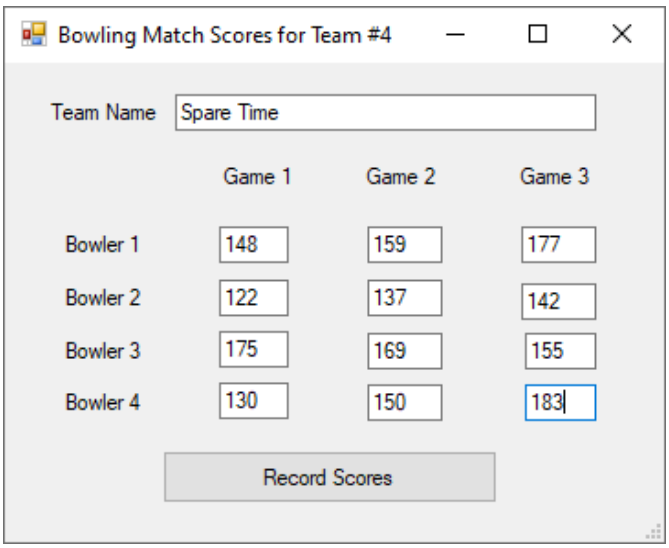
Secondary form used to enter the scores for one team

- Includes eight labels, 13 text boxes and one button.
- Form is used four times for data entry, once for each team.
- The form text value is updated after clicking the button to indicate the current team number for data entry.




Bowling Match Scores for Team #2
Team Name
High Rollers
Game 1
Game 2
Game 3
Bowler 1
181
193
176
Bowler 2
127
156
145
Bowler 3
198
211
300
Bowler 4
155
173
180
Record Scores


Bowling Match Scores for Team #3
Team Name
King Pins
Game 1
Game 2
Game 3
Bowler 1
225
226
207
Bowler 2
208
195
183
Bowler 3
199
188
201
Bowler 4
256
249
233
Record Scores


Bowling Match Scores for Team #4
Team Name
Spare Time
Game 1
Game 2
Game 3
Bowler 1
148
159
177
Bowler 2
122
137
142
Bowler 3
175
169
155
Bowler 4
130
150
183
Record Scores

Programming Project Suggestions

- The primary form will be used to store publicly accessible variables, a load event procedure that launches the secondary form, and a button click event that uses the scores array and names array data to identify the winning team, winning team score, and highest individual game score.
- The first publicly accessible variable is a three-dimensional array used to store the scores for the four teams, four bowlers per team, and three games per bowler. The array dimensions are 3, 3, and 2. The first dimension is the team (index 0-3), the second dimension is the bowler (index 0-3), and the third dimension is the score for one game (index 0-2). The scores array will be able to hold 48 game scores.
- The second publicly accessible variable is a one-dimension array that will store the four team names (index 0-3).
- The third publicly accessible variable is an integer that stores the current team number being processed in the second form.
- A load event procedure will launch the second form using a statement similar to `Form2.ShowDialog()`.
- The button click event procedure will identify the winning team, winning team score, and high individual game using the data in the arrays after they have been populated using the secondary score entry form. Example output is shown in the screen shots on the previous pages. A special message is displayed if someone bowls a 300, otherwise no special message will appear in the third line of the list box.
- Processing data in a three-dimensional array is typically done using three nested loops (index variables i, j and k). Several other variables will be needed to store the winning team and high score data so that it can be displayed after the loops are done processing all of the game scores.
- The secondary form will include a load event procedure that assigns the initial form text value “Bowling Match Scores for Team #1”.
- It will also include one other event procedure used each time the Record Scores button is clicked. This procedure will assign the team name and 12 scores to the appropriate part of the scores array. It will then clear each of the text boxes and put the focus back on the team name text box so that the next team’s scores can be entered.
- An example statement to assign one game score to the scores array is:
  `Form1.scores(Form1.teamNumber, 0, 0) = CInt(txtScore11.Text)`
- If the team number variable in the primary form is less than 3 (assuming team numbers are in the range of 0 to 3), then one will be added to the team number and the form title can be updated using the new team number. If four team’s scores have been entered, then the form can be closed using `Me.Close()`. This returns the focus back to the primary form so that the match results can be displayed.