

Project 5 – Rock Paper Scissors Simulation

You will create a program that allows users to collect and analyze data from playing multiple games of Rock, Paper, Scissors. Users can choose how many games the program will simulate (it doesn't take long to play 1,000,000 games with a simulator!) and can select from one of four player strategies (always rock, always paper, always scissors, or random). Users can also personalize the program by changing the player's name from the generic "player" to something else.

Rock smashes Scissors; Scissors cut Paper; Paper wraps Rock. You may use three other objects, but you must make the rules clear in your documentation.

Player Wins	Computer Wins	Ties
166,702	166,311	166,987
Player Win %	Computer Win %	Tie Win %
33.3 %	33.3 %	33.4 %

Rock %	Paper %	Scissors %
33.3 %	33.3 %	33.4 %

Objectives

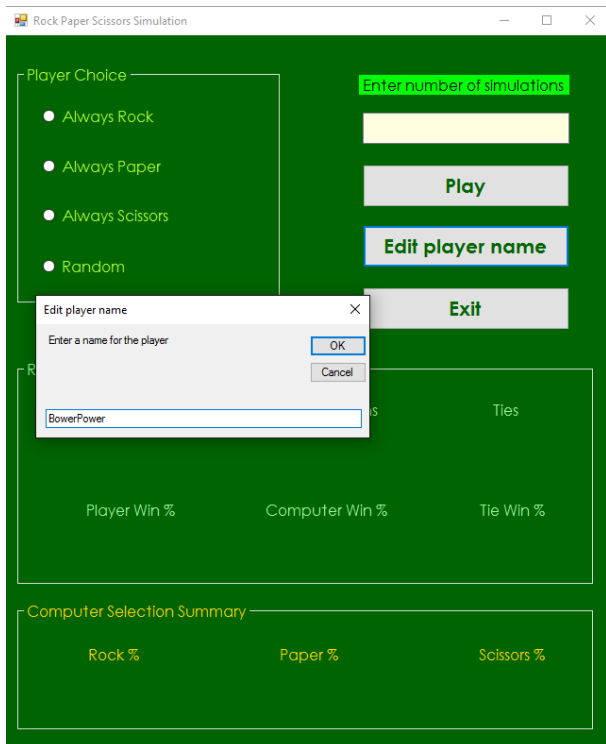
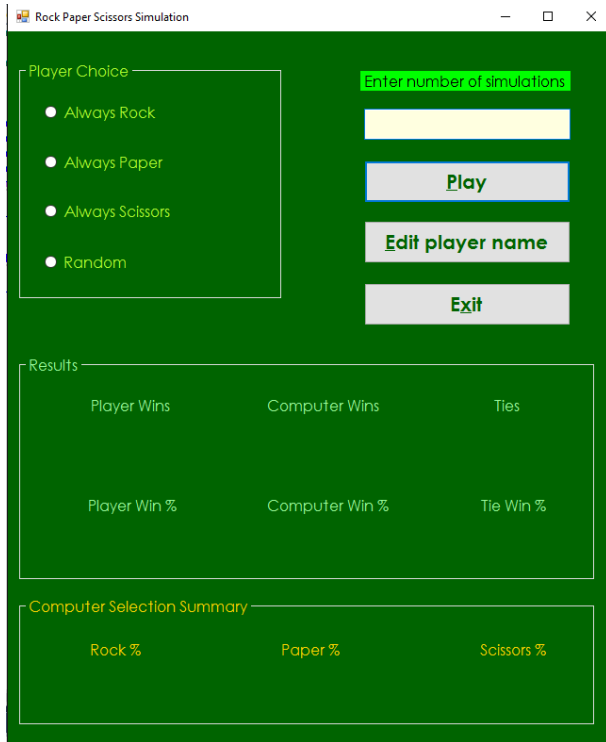
- Implement a **For** loop
- Use of an **InputBox**
- Use of **ToolTip**
- Use of accumulators (running totals) and counters
- Validate input
- Use of **Random**
- Obtaining data from an **InputBox**

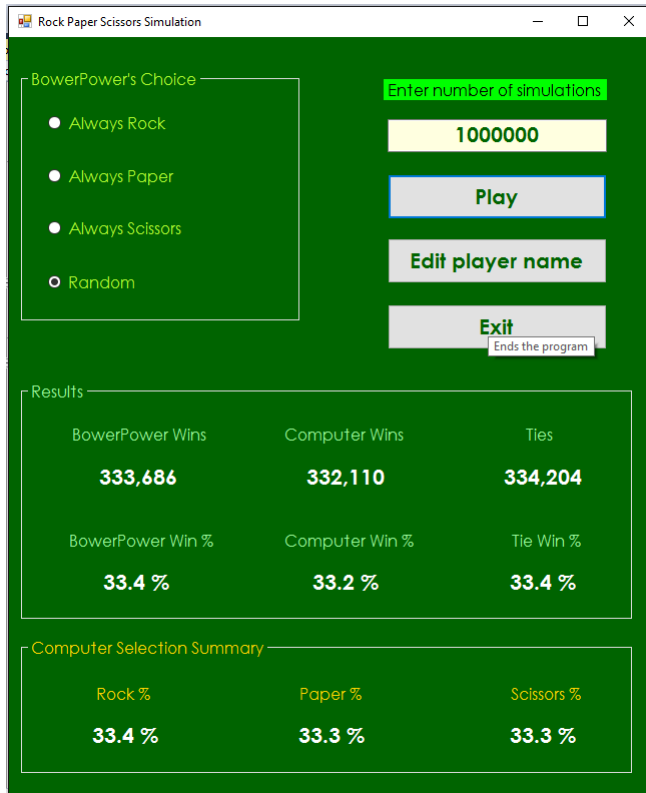
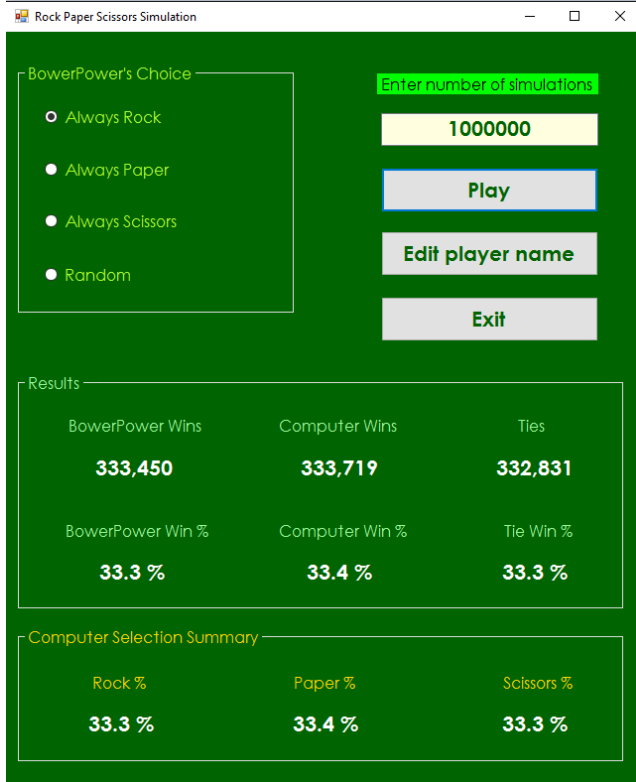
Requirements & Tips

- Comment your code!!!
- Make sure users type in a numeric value (use Try/Catch or TryParse) and that the value is greater than zero
- Radio buttons represent the four player strategies
 - always ROCK
 - always PAPER
 - always SCISSORS
 - random selection
- The game simulation begins when users click the PLAY button. Use a **For** loop to execute the games.
 - The computer's selection is always random
 - Use constants (declared with **Const**) to refer to the integers that stand for ROCK, PAPER, and SCISSORS - this will make your program much easier to read
- The program will calculate and display
 - the number of Player Wins
 - the percentage of the games the Player won
 - the number of Computer Wins
 - the percentage of the games the Computer won
 - the number of Ties
 - the percentage of games that were tied
- The program will also calculate and display
 - the percentage of the games the Computer selected ROCK
 - the percentage of the games the Computer selected PAPER
 - the percentage of the games the Computer selected SCISSORS
- All percentages should be formatted properly to one or two decimal place(s)
 - 34.7%, not 34.72815 or %34.7
- After users successfully use the EDIT PLAYER NAME, the word "Player" should change to the new player name in the labels and group box where it appears.
- Use a TextChanged event so clear all the results when users change the data in the textbox.
 - It is possible to additionally do this with a Click event for each radio button (or all the radio buttons in one sub) - this is not required, but you can learn more by watching a video about handling multiple events in one procedure:
 - Screencast (<http://screencast-o-matic.com/u/etC7/StateCapitalTest>)
 - YouTube (<https://youtu.be/Y9Wm-fn4zxw>)
 - Download the MP4 file (http://bowerpower.net/compprog1/vb06/HandlingMultipleEvents_OneEventHandler.mp4)
- The program ends when users click an EXIT button
- Tab order: the textbox where users type a number, the rest of your buttons (you select a logical order)
- Include keyboard access to the three buttons and tool tips for the buttons
- AcceptButton is PLAY; CancelButton is EXIT
- Use of an invisible radio button is optional (no player choice selected at start)

- Type and modify the following at the top of the code:
`'Your Name`
`'Date`
`'Assignment Name (Project 5 - Rock Paper Scissors, 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	7
Runs properly	6
Comments/documentation included	5
Use of constants to refer to rock, paper, & scissors	2
Edit Player Name button	2
Exit Button	1
Play Button	2
Use of random numbers	2
Validate user input using Try/Catch or TryParse	1
Formatting percentages	2
TextChanged event	1
For...Next	3
RadioButtons for four player strategies	2
Accurate results: Player wins, Computer wins, and Ties	3
Accurate results: Player win percentage, computer win percentage, tie percentage	3
Accurate results: Computer selection percentage	3
ToolTip	1
Run-time communication with user (includes use of InputBox)	4
TOTAL	50