Looping Assignment

Due date: ______________________

Objectives

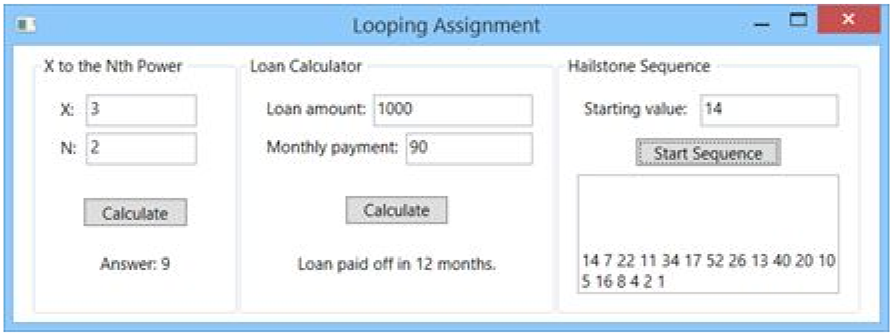
The purpose of this assignment is to assess your understanding of the following concepts:

- Looping statements
  - `while` (boolean expression) { … }
  - `do` { … } `while` (boolean expression)
  - `for` (initialization statement ; boolean expression ; iterator statement) { … }

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

This assignment is really three programs in one. Start with a Window layout that looks like the figure at right. The leftmost section has controls for program one (X to the Nth power). The middle section has controls for program two (loan calculator). The right section has controls for program three (hailstone sequence). Each of these section has a button that runs the corresponding program and a text block or read-only text box below the button that displays the final results of the program.



X: 3
N: 2
Answer: 9

Loan amount: 1000
Monthly payment: 90
Loan paid off in 12 months.

Starting value: 14
14 7 22 11 34 17 52 26 13 40 20 10
5 16 8 4 2 1

Each program should use a loop to make its calculation. <u>You must use a different type of loop (while, do..while, and for) for each program.</u> The following sections describe how each of the three programs work.

X to the Nth Power

This program takes two values as input (corresponding to the two text boxes): X and N. The program should calculate the value of x^n by multiplying a running total by x n times. (Hint: start the running total with a value of 1.) The program should then output the calculated value to the text block. For example, if the value of X is 2 and the value of N is 4 then the following would be written to the text block:

```
Answer: 16
```

Loan Payment Calculator

This program takes two values as input (corresponding to the two text boxes): loan amount and monthly payment. The program should calculate how many months it will take to pay off the loan by simulating paying off the loan over that many months. Each month, the remaining loan balance is decreased by the monthly payment amount. When the loan balance has reached zero the payoff is complete. The program should then output the number of months it took to pay off the loan in the text block.

For example, if the initial loan amount is $1,000 and the monthly payment is $200, the following would be written to the text block:

> Loan paid off in 5 months.

Hailstone Sequence

The hailstone sequence is defined as:

- If the current number N is even (N % 2 == 0), then the next number in the sequence is computed as N / 2.
- If the current number N is odd (N % 2 == 1), then the next number in the sequence is computed as N * 3 + 1.

The hailstone sequence will converge to 1 no matter what the starting value is (provided the starting value is an integer).

This program takes a single value as input (corresponding to the one text box): starting value. It then computes the hailstone sequence from the starting value down to 1, counting the number of cycles it takes to get there. The program should output the entire sequence to the text block or text box (the numbers can be separated by spaces).

For my program, I used a read-only text box to display the sequence. A read-only text box is a one with the `IsReadOnly` property set to true (checked). I also set the `VerticalScrollBarVisibility` property to `Auto`. Use the text box's `Clear` method to clear all text from the text box (at the beginning) and the `AppendText` method to write text to the end of the text box after whatever text is already there.

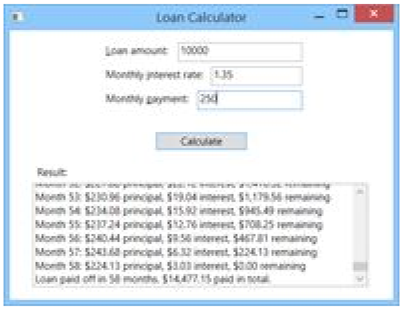
For example, for a starting value of 100, the following would be the sequence output to the text box (note that the starting value is included in the output):

```
100 50 25 76 38 19 58 29 88 44 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1
```

Points Breakdown

- (5 points) The program window contains all of the necessary controls (input edit boxes or numeric up-down controls for the input to each program, buttons to run each program, and a text block or text box for final output of each program.
- (10 points) The program uses meaningful variable and method names (including control names) that use the correct naming convention ("camelCase" for variables and "PascalCase" for methods).
- (25 points) The X-to-the-Nth-power program has a loop that calculates the correct value.
- (25 points) The loan payment calculator program has a loop that calculates the number of months it takes to pay off the loan.
- (25 points) The hailstone sequence program has a loop that calculates each number in the cycle.
- (10 points) Each program uses a different type of loop (while, do..while, and for).

Additional Learning



Loan amount: 10000

Monthly interest rate: 1.35

Monthly payment: 250

Result:

```
Month 53: $230.96 principal, $19.04 interest, $1,179.56 remaining
Month 54: $234.08 principal, $15.92 interest, $945.49 remaining
Month 55: $237.24 principal, $12.76 interest, $708.25 remaining
Month 56: $240.44 principal, $9.56 interest, $467.81 remaining
Month 57: $243.68 principal, $6.32 interest, $224.13 remaining
Month 58: $224.13 principal, $3.03 interest, $0.00 remaining
Loan paid off in 58 months, $14,477.15 paid in total.
```

- (1 ticket/5 bonus points) Change the hailstone program so it evaluates the sequence for every starting value from 1 to 1000 then outputs which starting value has the longest sequence.
- (1 ticket/5 bonus points) Is it possible for your loan payment calculator to have an infinite loop? (Yes, it is.) Change your program to detect if this happens and print an error in this case.
- (3 tickets/15 bonus points) Write a program that does the loan payment calculation using interest. In this case, each month before subtracting the monthly payment from the loan balance add an interest payment to the loan balance. The program should display how much from each monthly payment is going toward principal and interest, then at the end of the loop show how many months it took the pay off the loan and how much was paid in total. (Be sure not to overpay the last month.)