Homework 3

Computer Science I (20-CS-121) Summer 2007

Assigned: Wednesday, 27 June, 2007
Due: Friday, 5 July, 2007

1 Problem Description

For this homework, you'll be writing a *menu driven* program with three menu options. The options are

1. Approximate a value for e^x for a specific value of x and a given precision (the method is described below)
2. Approximate a value for π using the same method you did for lab 4
3. Exit the program

Please see lab 4 for the *Gregory-Leibniz* series and how to use it to calculate an approximation for π, if you have forgotten.

- **Approximating π:** As discussed in the 4^{th} lab assignment, one can represent π using the *Gregory-Leibniz* infinite series, which looks like the following...

$$\pi = 4\sum_{n=0}^{\infty}\frac{-1^n}{2n+1} = \frac{4}{1} - \frac{4}{3} + \frac{4}{5} - \frac{4}{7} + \frac{4}{9} - \frac{4}{11} + \dots \tag{1}$$

  Call each fraction to the right of the (second) equals sign above a *segment*. To approximate a value for π, you first prompt the user for the number of segments they want used in the approximation. Once you have that information, use it to actually calculate that much of the above series. The result will be your approximation.

- **Approximating e^x:** There is an infinite series that can be used to represent e^x, similar to the way we used the series above to represent (and approximate) π. The series for e^x is...

$$e^x = \sum_{n=0}^{\infty}\frac{x^n}{n!} = \frac{x^0}{0!} + \frac{x^1}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \dots \tag{2}$$

  Given a specific value for x, and a number of segments which to use in our approximation, you are to use the above series in the same way you did for approximating π. The only difference here (aside from the equation of course), is that you must also prompt the user for a value of x, which will be used in the approximation. *Note that for $x = 1$, your program should approximate e itself!*

2 Program Description

Your program should proceed as follows...

1. Display a menu similar to the following...

```
Please choose one of the following menu options...
1. Approximate Pi            2. Approximate e^x
                             3. Exit Program

Enter a Menu Selection:
```

2. If the user enters `3`, exit the program immediately
3. If the user enters `1`, prompt the user for the number of segments to use when approximating π, and then calculate the approximation and output it.

4. If the user enters `2`, prompt the user first for the value of x, then the value for the number of segments to use when approximating e^x. After you have that information, calculate the approximation and output it.

5. Once you finish approximating π or e^x, re-display the menu to the user, where they can once again select one of the 3 options

Your program *must* do the following...

- Since the number of segments used for approximating both π and e^x *must* be a whole number ≥ 1, if the user enters anything else you should repeatedly prompt them until they enter a correct number (and provide an appropriate message to the user why their value was incorrect).
- Compile without errors or warnings!
- Perform the specified task
- Employ a good design, *which you clearly document*
- Make use of good C++ style. Again, this does not have to be the style I covered in class...you can choose your own. *But whatever you use, you must be consistent!*

3 Questions to be Answered

Once you have the above program written, answer the following questions.

1. Report the approximations your program returns for π and e^1 (e^x where $x = 1$) when using 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 100, 200, and 500 segments.
2. What's the most accurate value of π your program can calculate?
3. What's the most accurate value of e^1 your program can calculate?
4. Is there anything you can do to your program to increase the accuracy when calculating π? If so, describe what you did.
5. Is there anything you can do to your program to increase the accuracy when calculating e^1? If so, describe what you did.
6. **Bonus (10 points).** When calculating your approximation for e^x for any x, if n segments are input, how many multiplications and additions are done? I.e. can you express the number of multiplications and additions done as some *function of* n? Could you reduce this number by exploiting the fact that $(n+1)! = (n+1) * n!$ (if you haven't already)?

4 How to submit your assignment

Once you are finished, you are to either

1. Email a copy of your source code and answers to the above questions to me at `ryan.flannery@gmail.com`
2. Submit a copy of your source code and answers through BlackBoard's Digital Drop-box

Either one must be done *before the start of class on Friday* and bring a print-out of your source code to class on Friday.