Programming Assignment #6

Monitor a Population

CS 1428.003 and 004, Fall 2015
Instructor: Jill Seaman

Due: in class Monday, 11/9/2015 (upload electronic copy by 10:00am)

Problem:

Your friend the biology lab assistant has a new job and now needs to monitor different populations of rabbits used in his lab. He needs to be able to predict the size of the population based on the initial population size and the average birth and death rates of the rabbits.

Write a program that asks the user for an initial population size, the birth rate and death rate, and the number of years to monitor the population. Then calculate and output a table of changes in population size over those years.

The formula to compute the new population is:

$$N = P + B * P - D * P$$

where N is the new population size, P is the previous population size, B is the birth rate, and D is the death rate. The birth and death rates are between 0 and 1.

Your program should output a table showing the starting population and the new population at the end of each year that the population is being monitored. To output the table, you will calculate the first year population using the initial population size for P, then calculate the second year using the first year population for P and so on, until you have calculated all of the requested years.

Finally, if the growth rate (birth rate - death rate) is positive you will notify the user in which year the population will double in size. The formula to calculate this is:

$$n = \log(2)/ \log(1 + r)$$

where `log` is a function in the cmath library, and r is the growth rate. If the growth rate is negative, you should notify the user in which year the population will become 0. The formula to calculate this is:

$$n = \log(1/P)/ \log(1 + r)$$

where P is the initial population size and r is the growth rate. If the growth rate is 0, you should notify the user that **the population is stable**.

After calculating these values, ask the user if they want to run the calculator again, and repeat as requested (accept 'y' or 'Y' as yes).

Input:

Ask the user for the initial population size, the birth rate, the death rate, and the number of years of population changes to report. The birth rate and death rate may have fractional values. You should validate the input as follows:
- The initial population size should be at least 100 but not more than 1,000,000.
- The birth rate and death rate should be greater than 0 and less than or equal to 100.
- The years should be greater than 0.

If the input is out of range, use an informative prompt to ask the user to re-enter valid data until the input is valid.

For the calculations to work properly, you must convert the birth and death rates which are input as a percent (i.e. 3.5%) to fractional values between 0 and 1 (i.e. 0.035).

Processing: You must use the following 5 functions in your program.

getInput: has four reference parameters to input initial population size, birth rate, death rate, and number of years from the user.

calculatePop: has 3 parameters: population size, birth rate, and death rate. Returns the calculated new population size. Do not do any output from this function.

yearsToDouble: has 1 parameter, the growth rate (which should be positive). Returns the calculated number of years it takes for the population to double in size. Do not do any output from this function.

yearsToZero: has 2 parameter, the initial population and the growth rate (which should be negative). Returns the calculated number of years it takes for the population to reach 0. Do not do any output from this function.

outputTable: has 4 parameters: initial population size, birth rate, death rate, and total years. It outputs the table with column headers, and one row for each year from 0 to total years, showing the population size at the end of that year.

Output: Display the years and population size as whole numbers. Use the function `floor` in the cmath library to output the population sizes as whole numbers. The values in the table should line up (right-justified). You may assume that the population size will fit within 20 characters. Use the function `ceil` in the cmath library to output the number of years to double and to reach 0 as whole numbers.

Sample output:

```
Enter the initial population: 500
Enter the annual birth rate (as % of current population): 15
Enter the annual death rate (as % of current population): 10
Enter the number of years of population changes: 20

Year          Population
----  ------------------
   0                 500
   1                 525
   2                 551
   3                 578
   4                 607
   5                 638
   6                 670
   7                 703
   8                 738
   9                 775
  10                 814
  11                 855
  12                 897
  13                 942
  14                 989
  15                1039
  16                1091
  17                1146
  18                1203
  19                1263
  20                1326

Population will double in 15 years.

Would you like to run the calculator again(Y/N)?: y

Enter the initial population: 100
Enter the annual birth rate (as % of current population): 1
Enter the annual death rate (as % of current population): 45
Enter the number of years of population changes: 10

Year          Population
----  ------------------
   0                 100
   1                  56
   2                  31
   3                  17
   4                   9
   5                   5
   6                   3
   7                   1
   8                   0
   9                   0
  10                   0
```

```
Population will be zero in 8 years.

Would you like to run the calculator again(Y/N)?: n
```

Additional Requirements:

- Your program **must compile** and run, otherwise you will receive a score of 0.

- Your program must pass **Test Case 0** (below) or you will receive a score of 50 or less with no credit for the other grading categories (correctness/constraints/style):

  Input: population size = 500, birth rate = 15, death rate = 10, years = 20.
  Expected output: see the first table in the sample output above. Years to double and option to run the calculator again are not required to pass Test Case 0 (they ARE required for a complete solution to the assignment). Your code must implement the two functions calculatePop and outputTable to pass Test Case 0.

- You may use integer or floating point data type to store the initial population size.

- Your program MUST define and call the functions described above.

Style:

See the Style Guidelines document on the course website. **Especially pay attention to the comments required for functions.** The grader will deduct points if your program violates the style guidelines.

Logistics:

Name your file **assign6_xxxxx.cpp** where xxxxx is your TX State NetID (your txstate.edu email id). The file name should look something like this: assign6_js236.cpp

There are two steps to the turn-in process:

1. Submit an electronic copy using the Assignments tool on the TRACS website for this class (tracs.txstate.edu). Submit the .cpp file, (NOT a .cbp file!).

2. Submit a printout of the source file at the beginning of class on the day the assignment is due. Please print your name on the front page, and staple if there is more than one page.

See the assignment turn-in policy on the course website (cs.txstate.edu/~js236/cs1428) for more details.