MQF Object Oriented Programming I Fall 2019
Hw1 Due 9/18/2019 before midnight

Specifications

Create a system that calculates future values giving interest rates. Your system contains a menu that list of four options.

The first option will calculate the future value of s sum of money. The formula is listed as the following

$$p_n = p_0(1+r)^n$$

Where p_n is the future value n period from now, p_0 is the original principal, and *r* is interest rate.

The second option will calculate future value of an ordinary annuity (payments are made at the end of the period). The formula is given as

$$p_n = A[\frac{(1+r)^n - 1}{r}]$$

Where *A* is the annuity amount and *n* is the number of periods and *r* is interest rate (rate of return or discount rate).

The third option will calculate future value of an annuity due (payments are made at beginning of the period). The formula is given as

$$p_n = A[\frac{(1+r)^n - 1}{r}](1+r)$$

Where *A* is the annuity amount and *n* and *r* are similar to the above.

When the fourth option is chosen, the system will quit and output a farewell message as “Thank you to use *your-name* future calculation system!”. Where *your-name* is your real name. For example, if your name is ***John Doe***, then option 4 will output **Thank you to use John Doe future calculation system!**

When user choose option 1, your program will ask the user to enter the original principal. Note principal and interest rate cannot be negative. If user choose option 2 or 3, your program will ask user to enter annuity, number of periods, and interest rate. All these numbers cannot be negative either.

After computation of either option 1, 2 or 3, your program will continue to output the main menu and ask user to make a selection. It will loop forever unless user selects 4 (for quit). If a user enters invalid integer such as 5, 100, etc., your system will continue to output the main menu and ask for a valid input (1, 2, 3, or 4). In our case, the valid input is an integer between 1 and 4.

Minimum Test Set **(follow this sequence, add other test cases, if you want, before your quit.)**

1 (first option)
1000 (original principal)

0.1 (interest rate)
2 (number of periods)

5 (invalid input)

2 (second option)
1000 (annuity amount)
0.1 (interest rate)
2 (number of periods)

2 (second option)
1000 (annuity amount)
0.05 (interest rate)
20 (number of periods)

3 (third option)
1000 (annuity amount)
0.1 (interest rate)
2 (number of periods)

-100 (invalid input)

4 (quit from system)

(Note on testing: when user choose option 1, 2, and 3. you might test your system with negative numbers as input for principal (or annuity), period and interest rate. in these cases, your system shall continue to ask user to input until user enter valid inputs. It will be counted as if your system run correctly)

Requirements

(1) Comment your program	3
(2) Meaningful variable names	2
(3) Appropriate indentation	2
(4) Use (at least) testing set provided	3
(5) Correct results	10

Submission

(1) Create a folder and put all your source code file or files (.cpp file, .h) into a folder. Use *yourLastName-FirstName-Hw1* as the name of the folder. E.g. if your name is *Kathy Smith* then *Smith-Kathy-hw1* is the folder name

(2) Create a document (e.g. a .docx Word file) and copy and paste your source code files(s) into this document. Name this file as *Smith-Kathy-hw1-report* assume your name is Kathy Smith.
(3) Test run your program and record the output by copy the screen shots.
(4) Copy and paste all your screen shots into the document you created at step(2)
(5) Convert this file at step(4) into a .pdf file if possible
(6) Put the file above (step(5)) into the folder created at step(1)
(7) Zip the folder and Use yourLastName-FirstName-Hw1 as the zipped filename.
(8) Submit the zipped file to blackboard.

Note:

(1) Independent work, not team project. Discussion is allowed, but no copying.
(2) You may use more test set. But test the testing set provided first based on the order given
(3) Please send questions and comments to your instructor and TA.
(4) If you are using Microsoft Visual C++, choose console application or create project using “Empty project” rather than include precompiled headers.

References:

Annuity and annuity due: https://www.investopedia.com/terms/f/future-value-annuity.asp