Program 1: Simple Mortgage Calculator...

Due date: Monday, January 31, 11:59pm.

1 Purpose

To write a program requiring use of variable declarations, reading values into and writing the values stored in variables, assignment statements and conditional statements.

Programming environment. This is a solo programming project. You are responsible for all the work related to the program development, testing and submission.

Collaboration. **Any** collaboration between peers, as well as any collaboration with outside sources is **strictly prohibited**. If you have any questions, concerning the assignment, please consult the instructor.

2 Program Description

You are writing a simple mortgage calculator. The user of your program is planning to purchase a house and wants to find out how much the payment would be. The user supplies the program with the terms of the mortgage, and your program computes the overall amount of money the user will owe the bank over time, the monthly payment, the annual payment, and a few other things.

A formal description of the program is given below.

Program Requirements.

TR0. Economics. Recall from high school how the mortgage-like financial instruments operate.

Parameter name	Notation	Explanation
House price	h	price of the house
Principal	p	amount of money borrowed
Interest Rate	r	% of principal to be repaid each year
Term	t	number of years for the loan

Using this information, the monthly payment on the loan is calculated as follows. First, we compute the monthly rate:

$$mr = \frac{r}{12}.$$

The total number of payments on the loan is:

$$N = 12 * t.$$

Now, the monthly payment is:

$$mPayment = \frac{p \cdot mr}{1 - (1 + mr)^{-N}}.$$

The annual payment is:

$$aPayment = mPayment \cdot 12.$$

The rest of the computations you will perform in the program are going to be straightforward.

The mortgage calculator will only compute the terms of a standard fixed-interest rate loan.

TR1. Overview. Your program will consist of two parts: mortgage calculator part and analysis part.

In the mortgage calculator part, your program will read a number of inputs, defining the mortgage parameters and will calculate the monthly, annual, total mortgage amounts as well as a few other important pieces of data, and report them to the user.

In the analysis part your program will analyze the resulting mortgage and will provide some feedback based on the observed information.

TR2. Mortgage calculator part overview. The mortgage calculator proper part of the program inputs the mortgage parameters, the repayment information. There are four parameters to be entered by te user:

1. House price
2. Loan amount
3. Interest rate
4. Loan term

TR3. Program data. Your program shall have a number of declared variables to represent all information computed by the mortgage calculator. All input variables are `float` except for loan term, which is an `int`.

All variables you use in computation shall be `double` except for the variable(s) representing the total number of payments over the course of the mortgage,which shall be `int`.

TR4. Units of measurement. Your program shall use the following units of measurement:

Parameter	Unit of measurement	Example
House price	*Dollars*	5000000.00
Loan amount	*Dollars*	4000000.00
Interest reate	*per cent*	5.5
Loan term	*Years*	30

T5. Output. All output of your program **must match exactly** the output of instructor's program given to you. This includes newlines, extra or missing spaces, and any extra or missing punctuation.

TR6. Data entry. Your program shall start by printing out a simple greeting message:

```
Mortgage Calculator
```

and skipping a line.

All lines that are not input prompts in the program shall end with newline character. Input prompts shall not contain newline characters.

After the greeting, the program shall print:

```
Enter price of the house: 
```

(ends with a space).

The program shall read the price of the house in dollars. After that, the program shall print:

```
Enter the amount you borrow: 
```

(ends with a space)

The program shall read the amount of money, in dollars, that a user wants to borrow. On the following line, the program shall print:

```
Enter the interest rate: 
```

(ends with a space)

The program shall read the interest rate in per cent out of 100. After that, the program shall print

```
Enter the term of the mortgage in years: 
```

(ends with a space)

The program shall read the term of the loan in full years.

TR7. Calculations.

Following the data entry, your program shall compute the following data:

- The down payment (difference between the house price and the loan amount).
- The share of the downpayment to the full price of the house (in per cents out of 100).
- The monthly interest rate (annual interest rate divided by the number of months in a year).
- Total number of payments.
- The monthly payment on the loan.
- The annual payment on the loan.
- Total amount of money needed to repay the loan.
- Total interest payment (total amount of money to repay the loan, minus the amount of the loan principal).
- Total amount of money the user will spend on the house (including the down payment).
- The ratio of the total amount of interest payments to the loan principal as a per cent out of 100.

Use requirement **TR0.** for the math behind these computations. Any computations not specified there should be straightforward.

TR8. Mortgage Calculator Report. The program shall skip one line, and then print:

```
------------- Report ----------------
```

(make certain you use the correct number of dashes)

After that, your program will output the following lines:

1. The first line:

   `Monthly payment: $`

   followed by the amount of the monthly payment.
2. The second line:

   `Annual payment: $`

   followed by the amount of the annual payment.
3. The third line:

   `Total amount to repay: $`

   followed by the amount the user would have to pay the bank.

4. The fourth line:

   `Total interest payments: $`

   followed by total amount of the interest payments.

5. The fifth line:

   `Interest is XX.XX% of the principal`

   where you shall insert the actual percentage you compute in place of `XX.XX`. (note the "%" character in the output).

6. Leave the sixth line empty.

7. The seventh line:

   `Down payment: $`

   followed by the amount of the down payment.

8. The eightth line:

   `Total spent on the house: $`

   followed by the total amount the user will spend on the house.

End the report with an empty line, followed by the line shown below:

```
--------------------------------------
```

All numbers should be reported as floating point numbers with two decimal places (payments are always made in terms of dollars and cents).

TR9. Analysis overview. The analysis part of the program, will attempt to provide an advice on whether the user should take the mortgage. It will ask for a few additional inputs, and then provide some feedback based on all the information supplied to the program and computed in it.

TR10. Extra information. Your program shall output an empty line, then print

```
-------------- Analysis -----------------
```

then output another empty line and request the following two extra pieces of information:

1. The square footage of the house. The prompt shall be

   `Enter the square footage of the house:`

   (ends with a space)

2. The average rent amount (in dollars) in the user's area. The prompt shall be:

   `Enter the average monthly rent in your area:`

   (ends with a space)

Both values are of type `float`.

TR11. Analysis. Computations.

Based on the new information, your program shall compute the following quantities:

- Price per square foot of the house (based on the price of the house).
- The total amount of rent one would pay over the period of the mortgage.
- The ratio between the total cost of the house and the total amount of rent over the period of the mortgage.

TR12. Analysis. Price of the house.

Based on the computed price of the square foot of the house, your program will render an opinion on how expensive the house is.

If the price per square foot is:

Greater or equal than	but less than	Output
0	100	`The house is relatively inexpensive`
100	200	`The house price is reasonable`
200	300	`The house price is above average`
300	∞	`The house is very expensive!`

TR13. Analysis. Should user buy? Based on the ratio between the total cost of the house, and the total cost of rent, the program shall render a purchasing advise according to the following table:

If the ratio is:

Greater or equal than	but less than	Output
0	1	`Buying a house is cheaper than renting!  Go for it!`
1	1.25	`Buying a house costs about the same as renting.  Good deal!`
1.25	2	`Buying a house is more expensive than renting.  You need a good reason`
2	∞	`Renting is a MUCH better option than buying this house.  Don't do it!`

This shall complete the work of your program.

TR15. Program name. Name your program `mortgage.c`.

TR16. Input guards. You are guaranteed that all entered data will be valid.

General Notes

Math. You are responsible for the remainder of the program design for this program. In particular, you are responsible for coming up with the correct math to compute the outputs of the program based on the inputs.

ANSI C. Your program shall be written in ANSI C. The instructor will compile your program using the following `gcc` flags:

```
gcc -ansi -Wall -Werror -lm
```

(note that you may need to use `pow()` function from the `math.h` library package, hence `-lm` flag may be needed when you compile.)

Any program that does not compile in this fashion will be assigned a score of 0.

Style. Your code will checked for style. Your program shall conform to the style described at

http://users.csc.calpoly.edu/~cstaley/General/CStyle.htm

In addition, the header comment shall be as described in Lab 2 specification.

Any style violations are subject to an automatic 10% penalty.

Testing. You are be provided with the instructor's binary, `mortgage-alex` and with a set of test cases for testing your program. We also include two standard testing scripts, `mortgage-tests.csh` for your program and `mortgage-alex-tests.csh` for the instructor's binary.

Please note, that you are expected to use both the instructor's binary, **and** the test cases and test scripts in your work.

Outputs of instructor's and your program **must match**. The only dispensation is given to rounding errors due to floating point computations.

Any program that fails any of the public tests will not receive more than 30% of the grade.

3 Submission Instructions

Submission.

Files to submit. You shall submit the `mortgage.c` file.

No other files shall be submitted.

Submission procedure. You will be using `handin` program to submit your work. The command is as follows:

for Section 01 students

```
> handin dekhtyar-grader program01-01 mortgage.c
```

for Section 09 students

```
> handin dekhtyar-grader program01-09 mortgage.c
```

Late submission. You may submit late for a 24-hour period following the deadline. Late submissions are subject to the standard 10%—30% penalty at the instructor's discretion.

4 Sample Output

```
$ mortgage-alex
Mortgage Calculator

Enter price of the house: 500000
Enter the amount you borrow: 400000
```

```
Enter the interest rate: 5.5
Enter the term of the mortgage in years: 30

-------------- Report -----------------
Monthly payment: $2271.16
Annual payment:  $27253.87
Total amount to repay:   $817616.14
Total interest payments: $417616.14
Interest is 104.40% of the principal

Down payment: $100000.00
Total spent on the house: $917616.14

--------------------------------------

-------------- Analysis -----------------

Enter the square footage of the house: 2000
Enter the average monthly rent in your area: 2000
The house price is above average
Buying a house is more expensive than renting. You need a good reason
```