Project 5d: Loan payoff

When you borrow money to buy a house, a car, or for some other purpose, you repay the loan by making periodic payments over a certain time. Of course the lending company will charge interest on the loan. Every periodic payment consists of the interest on the loan and the payment toward the

principal amount. To be specific, suppose that you borrow $1000 at the interest rate of 7.2% (0.072 as a decimal) per year and the payments are monthly. Suppose that your monthly payment is $25. The interest is 7.2% per year and the payments are monthly, so the interest rate per month is 7.2 / 12 or 0.06% (0.006 as a decimal. The first month’s interest on $1000 is 1000 \* 0.006 or $6.00 Because the payment is $25 and interest for the first month is $6, the payment toward the principal amount is 25 – 6 or $19. This means that after making the first payment, the loan amount is 1000 – 19 or $981. For the second payment, the interest is calculated on $981. So the interest for the second month is 981 \* 0.006 or 5.886, that is, approximately $5.89. This implies that the payment toward the principal is 25 – 5.89 = 19.11, and the remaining balance after the second payment is 981 – 19.11 or 961.89. This process is repeated until the loan is paid. Write a program that accepts as input the loan amount, the interest rate per year, and the monthly payment. (Enter the interest rate as a percentage. For example, if the interest rate is 7.2% per year, then enter 7.2) The program then outputs the number of months it would take to repay the loan. (Note: If the monthly payment is less than the first month’s interest, then after each payment, the loan amount will increase. In this case, the program must warn the borrower that the monthly payment is too low and with this monthly payment the loan amount could not be repaid.)

Allow the user to evaluate as many loan / repayment setups as they like and allow them to quit any time they desire by entering a particular value such as “quit” any time they are prompted for an input. Each input must be validated allowing the user to correct invalid entries or quit, if they so choose. The number years must be greater than 0 and a whole number. The interest rate must be in the range of 5.6% and 10%. The monthly payment must be greater than zero. Additionally make sure and warn the user if the monthly payment is too low to allow the loan to be repaid in the term chosen and allow them to change their proposed monthly payment.

Evaluation:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Aspect | Objectives substantially met  90 – 100% | Meets Minimal Requirements  80-89% | Needs Improvement  79 – 79% | Failure to Meet Requirements  0 - 69% |
| Good Programming Practices  10% | Effective use of white space.  Clear and appropriate documentation.  Implements all needed error handling.  Data types and identifiers meet all expectations. | Use of white space and documentation with minor defect. Implementation of error handling general but lacking in minor aspects.  Selection of data types and / or creation of identifiers not consistent. | Generally meets expectations for good programming practices. | In the main does not meet expectations for good programming practices. |
| Selection Structures  20% | All selection structure of the best type for problem as posed. | Generally the best structure is implemented. | Some selections structures are appropriate. | In the main the selection structures do not fulfill the specifications. |
| Iterative Structures  30% | All iterative structures appropriate to the algorithm proposed.  Control variables correctly formed and modified for program control |  |  |  |
| Accuracy of Output  40% | All output correct and supported with appropriate testing. | With minor exceptions the output is correct and testing may be missing some needed test cases. | Only part of the output is correct as a result of inadequate test cases. | Only some of the output is correct as a result of missing or misused test cases. |