

CS 1014: Numerical Computational Techniques

Project 4

Due April 9, 1999

LOAN AMORTIZATION

All work for this project must be done by you alone. You are not allowed to receive any help from anyone except the course instructor or the TA.

This assignment will give you the opportunity to 1) use the repetition construct DO WHILE . . . END DO (or DO . . . END DO), 2) use the selection construct IF . . . THEN . . . ELSE, . . . END IF, 3) practice formatting output, and 4) use the four basic arithmetic operations of addition, subtraction, multiplication, and division.

Problem Statement:

Example on loan amortization with fixed interest rate:

Suppose you take out a loan of \$1,000 at a yearly interest rate of 12% and you intend to repay that loan at \$200 (including the interest payment) per month until the loan has been fully repaid. You would like to have an amortization schedule showing the following entries: the interest charge, the payment towards the loan, the remaining balance, and the accumulated interest to date. One way to solve this problem is shown here (the monthly interest rate is 1% since the yearly rate is 12%, i.e., 12%/12 months = 1% per month):

The interest for month 1 is	$\$1000 * 0.01$	= \$ 10.00
The payment to capital is	$\$200 - 10$	= \$190.00
The remaining balance is	$\$1000 - 190$	= \$810.00
The interest for month 2 is	$\$810 * 0.01$	= \$ 8.10
The payment to capital is	$\$200 - 8.10$	= \$191.90
The remaining balance is	$\$810 - 191.90$	= \$618.10

:

---- The process continues until the balance is zero ----

Problem:

In this project, you are required to write a program for a bank to produce amortization schedules for loans to be paid by its customers according to the following (unusual) policies:

1. A customer can pay the loan in three different time segments.
2. Except the third time segment, each time segment is bounded by a fixed number of years. In the third time segment, payments to the loan should continue indefinitely until it is paid entirely.
3. The interest rate is fixed in a time segment but varies from one time segment to another.
4. The monthly payment is fixed in a time segment but varies from one time segment to another.

In contrast with the previous example, the problem deals with variable interest rates and monthly payments as far as time segments are concerned.

Notice that in some situations, it may be possible for a customer to finish all payments for the loan in one time segment or in two time segments since the schedule depends on the interest rate, the monthly payment, and the duration of each time segment.

Input:

The input file called "**loan.dat**" contains four lines of data.

Line 1: Loan amount
 Line 2: Monthly payment amount, yearly interest rate, and number of years for the first time segment
 Line 3: Monthly payment amount, yearly interest rate, and number of years for the second time segment
 Line 4: Monthly payment amount and yearly interest rate for the last time segment

Notice that for a yearly interest rate of 0.115 signifies 11.5% per year. Except the number of years (INTEGER), all other items are REAL. **Also note that** you do not need a READ loop in your program since the program requires to read all four lines of data once. Use the following data set to test your program:

Output:

Your program should create and send output to a file named "**report4.out**". Your output should have the exact form illustrated below. It has one blank line between the title line and the column headers and all words are in upper case. Failure to follow this form exactly will result in the automated grader deducting points. Be careful about the output line for the last month in which case the balance becomes zero after paying the amount on the remaining balance from the previous month and the interest on it. To test your program thoroughly for all three time segments, you should run your program with the following test data sets. For each test case, you need to modify the input file "loan.dat" first with the data set and then run your program to check the output file "report4.dat".

1) Test Case 1: Payments end in the first time segment

Data Set:

5000.00		
300.00	0.10	3
250.00	0.12	2
200.00	0.18	

Program Output:

AMORTIZATION TABLE

PAYMENT NUMBER	BALANCE	PERCENT RATE	MONTHLY PAYMENT	INTEREST PAID	AMORTIZATION	REMAINING BALANCE
1	5000.00	10.00	300.00	41.67	258.33	4741.67
2	4741.67	10.00	300.00	39.51	260.49	4481.18
3	4481.18	10.00	300.00	37.34	262.66	4218.52
4	4218.52	10.00	300.00	35.15	264.85	3953.68
5	3953.68	10.00	300.00	32.95	267.05	3686.63
6	3686.63	10.00	300.00	30.72	269.28	3417.35
7	3417.35	10.00	300.00	28.48	271.52	3145.82
8	3145.82	10.00	300.00	26.22	273.78	2872.04
9	2872.04	10.00	300.00	23.93	276.07	2595.97
10	2595.97	10.00	300.00	21.63	278.37	2317.61
11	2317.61	10.00	300.00	19.31	280.69	2036.92
12	2036.92	10.00	300.00	16.97	283.03	1753.89
13	1753.89	10.00	300.00	14.62	285.38	1468.51
14	1468.51	10.00	300.00	12.24	287.76	1180.75
15	1180.75	10.00	300.00	9.84	290.16	890.59
16	890.59	10.00	300.00	7.42	292.58	598.01
17	598.01	10.00	300.00	4.98	295.02	302.99
18	302.99	10.00	300.00	2.52	297.48	5.52
19	5.52	10.00	5.56	0.05	5.52	0.00

2) Test Case 2: Payments end in the second time segment

Data Set:

5000.00		
300.00	0.10	1
250.00	0.12	2
200.00	0.18	

Program Output:

AMORTIZATION TABLE

PAYMENT NUMBER	BALANCE	PERCENT RATE	MONTHLY PAYMENT	INTEREST PAID	AMORTIZATION	REMAINING BALANCE
1	5000.00	10.00	300.00	41.67	258.33	4741.67
2	4741.67	10.00	300.00	39.51	260.49	4481.18
3	4481.18	10.00	300.00	37.34	262.66	4218.52
4	4218.52	10.00	300.00	35.15	264.85	3953.68
5	3953.68	10.00	300.00	32.95	267.05	3686.63
6	3686.63	10.00	300.00	30.72	269.28	3417.35
7	3417.35	10.00	300.00	28.48	271.52	3145.82
8	3145.82	10.00	300.00	26.22	273.78	2872.04
9	2872.04	10.00	300.00	23.93	276.07	2595.97
10	2595.97	10.00	300.00	21.63	278.37	2317.61
11	2317.61	10.00	300.00	19.31	280.69	2036.92
12	2036.92	10.00	300.00	16.97	283.03	1753.89
13	1753.89	12.00	250.00	17.54	232.46	1521.43
14	1521.43	12.00	250.00	15.21	234.79	1286.65
15	1286.65	12.00	250.00	12.87	237.13	1049.51
16	1049.51	12.00	250.00	10.50	239.50	810.01
17	810.01	12.00	250.00	8.10	241.90	568.11
18	568.11	12.00	250.00	5.68	244.32	323.79
19	323.79	12.00	250.00	3.24	246.76	77.03
20	77.03	12.00	77.80	0.77	77.03	0.00

3) Test Case 3: Payments end in the third time segment

Data Set:

5000.00		
200.00	0.12	1
180.00	0.15	1
150.00	0.18	

Program Output:

AMORTIZATION TABLE

PAYMENT NUMBER	BALANCE	PERCENT RATE	MONTHLY PAYMENT	INTEREST PAID	AMORTIZATION	REMAINING BALANCE
1	5000.00	12.00	200.00	50.00	150.00	4850.00
2	4850.00	12.00	200.00	48.50	151.50	4698.50
3	4698.50	12.00	200.00	46.99	153.01	4545.48
4	4545.48	12.00	200.00	45.45	154.55	4390.94
5	4390.94	12.00	200.00	43.91	156.09	4234.85
6	4234.85	12.00	200.00	42.35	157.65	4077.20
7	4077.20	12.00	200.00	40.77	159.23	3917.97
8	3917.97	12.00	200.00	39.18	160.82	3757.15
9	3757.15	12.00	200.00	37.57	162.43	3594.72
10	3594.72	12.00	200.00	35.95	164.05	3430.67

11	3430.67	12.00	200.00	34.31	165.69	3264.97
12	3264.97	12.00	200.00	32.65	167.35	3097.62
13	3097.62	15.00	180.00	38.72	141.28	2956.34
14	2956.34	15.00	180.00	36.95	143.05	2813.30
15	2813.30	15.00	180.00	35.17	144.83	2668.47
16	2668.47	15.00	180.00	33.36	146.64	2521.82
17	2521.82	15.00	180.00	31.52	148.48	2373.34
18	2373.34	15.00	180.00	29.67	150.33	2223.01
19	2223.01	15.00	180.00	27.79	152.21	2070.80
20	2070.80	15.00	180.00	25.88	154.12	1916.68
21	1916.68	15.00	180.00	23.96	156.04	1760.64
22	1760.64	15.00	180.00	22.01	157.99	1602.65
23	1602.65	15.00	180.00	20.03	159.97	1442.68
24	1442.68	15.00	180.00	18.03	161.97	1280.72
25	1280.72	18.00	150.00	19.21	130.79	1149.93
26	1149.93	18.00	150.00	17.25	132.75	1017.18
27	1017.18	18.00	150.00	15.26	134.74	882.43
28	882.43	18.00	150.00	13.24	136.76	745.67
29	745.67	18.00	150.00	11.19	138.81	606.85
30	606.85	18.00	150.00	9.10	140.90	465.96
31	465.96	18.00	150.00	6.99	143.01	322.95
32	322.95	18.00	150.00	4.84	145.16	177.79
33	177.79	18.00	150.00	2.67	147.33	30.46
34	30.46	18.00	30.92	0.46	30.46	0.00

Documentation:

Do not forget to document your program. Some of your programs will be randomly selected from the archive (maintained on the grader server) later on for manual grading. Points will be deducted if your program is not well documented, well-structured, and readable. You can use the program handed out for the first project as a model for documentation. Do not forget to include your name(s) in the program documentation.

Submission:

You should submit your source code electronically to the automated grader. Before you submit your program, make sure your program runs perfectly and generates right results with some test input data. Your program will not receive any input from the keyboard or will not display any results on the screen. Do not submit the input or the output file. You will be allowed a maximum of four submits for this project. Remember, the late penalty is 20% for each day late.