

FACULDADE DE TECNOLOGIA DE SÃO PAULO
ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

GABRIEL DOS SANTOS CAMILO RA: 21110149

GABRIEL SOARES DA SILVA GOES RA: 21120011

GUILHERME VAIANO NOGUEIRA MENDONÇA RA: 21110013

PROJETO: SISTEMAS DE AMORTIZAÇÃO

PROGRAMAÇÃO ORIENTADA A OBJETO

(ILP-007)

SÃO PAULO, SP

2022

SUMÁRIO

| | | |
|-----|---|----|
| 1 | INTRODUÇÃO | 3 |
| 2 | ELABORAÇÃO DO PROJETO | 4 |
| 2.1 | Sistema de Amortização Constante (S.A.C.) | 4 |
| 2.2 | Sistema de Amortização Americano (S.A.A.) | 4 |
| 2.3 | Sistema de Amortização Francês | 5 |
| 2.4 | Sistema de Amortização Alemão (S.A.ALM) | 5 |
| 2.5 | Sistema de Amortização Misto (S.A.M.) | 6 |
| 3 | CRIAÇÃO DA CLASSE | 8 |
| 4 | TESTES DO PROGRAMA | 12 |
| 4.1 | Teste do Método S.A.C. | 12 |
| 4.2 | Teste do método S.A.F. | 12 |
| 4.3 | Teste do método de amortização americano | 13 |
| 4.4 | Teste do método S.A.M. | 14 |
| | REFERÊNCIAS | 15 |

1 INTRODUÇÃO

O presente documento tem como objetivo apresentar o desenvolvimento do projeto 2 da disciplina Programação Orientada a Objetos (ILP-007) do quarto semestre de Análise e Desenvolvimento de Sistemas da Fatec São Paulo, lecionada pelo professor José Paulo Ciscato.

No decorrer do documento serão apresentadas as bases utilizadas para desenvolver o programa, assim como as fórmulas utilizadas para a criação do algoritmo. Também será exposto o código do sistema para visualização e entendimento do mesmo.

2 ELABORAÇÃO DO PROJETO

A ideia do projeto consiste em desenvolver uma classe em Java que realize o cálculo de um empréstimo, financiamento ou dívida a partir de um dos sistemas de amortizações existentes.

O projeto foi elaborado a partir do que foi apresentado na disciplina do quarto semestre de Economia e Finanças (sigla CEF-100) no curso de Análise e Desenvolvimento de Sistemas da Fatec-SP. A seguir, serão apresentados os sistemas de amortizações que foram utilizados como base para a criação da classe e seus respectivos métodos.

2.1 Sistema de Amortização Constante (S.A.C.)

O Sistema de Amortização Constante (também chamado de Método Hamburguês) consiste na amortização constante da dívida com base em pagamentos periódicos decrescentes. Ou seja, quanto mais o tempo passa, menores ficam as parcelas de quitação do saldo devedor enquanto o valor é amortizado de maneira constante em todos os períodos.

De forma geral, os juros e o capital são calculados uma única vez e divididos para o pagamento em várias parcelas durante o prazo de quitação. O valor da amortização é calculado através da divisão entre o capital inicial e o número de prestações a serem pagas. O cálculo é feito da seguinte maneira:

$$A = P / n$$

Onde:

- **A** = Amortização;
- **P** = Principal;
- **n** = Número de prestações.

2.2 Sistema de Amortização Americano (S.A.A.)

Nesse sistema, a devolução do principal é feita de uma só vez no final do período de amortização. Não são previstas amortizações intermediárias durante

a vigência dos pagamentos, sendo os juros pagos periodicamente. Devido ao impacto financeiro que o próprio sistema concebe visando o pagamento único da dívida, faz-se necessária a constituição de um fundo de amortização, o qual irá financiar tal quantia. O fundo é constituído concomitantemente aos pagamentos de juros do principal através do uso do fator de acumulação de capital por operação múltipla.

2.3 Sistema de Amortização Francês

O Sistema de Amortização Francês, também conhecido como amortização por Tabela *Price*, é um dos modelos mais conhecidos, através dele, o montante total é amortizado ao longo do contrato e de forma crescente. Assim, o pagamento é feito através de um conjunto de prestações sucessivas e constantes.

Geralmente, as parcelas são pagas mensalmente em valores iguais, já com os juros embutidos. O cálculo é feito da seguinte maneira:

$$R = P * (((1 + i)^n * i) / ((1 + i)^n - 1))$$

Onde:

- **R** = Prestação;
- **P** = Principal;
- **i** = Taxa de juros;
- **n** = Número de prestações.

2.4 Sistema de Amortização Alemão (S.A.ALM)

O sistema Alemão consiste em liquidar uma dívida onde os juros são pagos antecipadamente com prestações iguais, exceto o primeiro pagamento que corresponde aos juros cobrados no momento da operação financeira.

Nesse sistema, a última amortização (A_n) deve coincidir com o pagamento P_n , uma vez que todos os juros são cobrados antecipadamente nas prestações anteriores, assim como todos os pagamentos, com exceção do primeiro, devem ser iguais. O cálculo desse sistemas utiliza as seguintes fórmulas:

- **Fórmula utilizada para calcular o valor de cada prestação:**

$$P = \frac{C \cdot i}{1 - (1 - i)^n}$$

- **Fórmula utilizada para calcular a primeira amortização:**

$$A_1 = P \cdot (1 - i)^{n-1}$$

- **Fórmula utilizada para calcular cada amortização após a primeira:**

$$A_k = \frac{A_1}{(1 - i)^{k-1}}$$

Onde:

- **C** = Capital financiado;
- **i** = Taxa de juros ao período;
- **n** = Número de períodos;
- **P** = Valor de cada prestação.

2.5 Sistema de Amortização Misto (S.A.M.)

O Sistema de Amortização Misto (SAM) é um sistema de amortização que apresenta características intermediárias entre os outros dois principais sistemas

de amortização, o Sistema de Amortização Constante (SAC) e o Sistema de Amortização Francês (SAF).

Logo, no sistema de amortização misto, o tomador do empréstimo irá pagar uma parcela que é a média entre o SAC e o Sistema *Price*. Para realizar esse cálculo, é necessário somar a parcela do SAC e do Sistema *Price* e dividir por dois. Portanto, pode-se dizer que as parcelas do SAM possuem uma lógica intermediária entre os dois sistemas.

3 CRIAÇÃO DA CLASSE

Após obter o conhecimento sobre como funcionam os sistemas de amortização, deu-se início ao desenvolvimento da classe Amortização. A seguir, será apresentado o código da classe que desenvolvido na linguagem Java através do ambiente de desenvolvimento Eclipse e Visual Studio Code.

```

1 package principal;
2 import java.math.RoundingMode;
3 import java.text.DecimalFormat;
4 import java.util.Locale;
5 import java.util.Scanner;
6
7 public class Amortizacao {
8     double periodo, saldoatual, taxajuros;
9
10    // Construtor inicializador
11    public Amortizacao (double pr, double sa, double txj) {
12        periodo = pr;
13        saldoatual = sa;
14        taxajuros = txj;
15    }
16
17    // Construtor cópia
18    public Amortizacao(Amortizacao A) {
19        this.periodo = A.periodo;
20        this.saldoatual = A.saldoatual;
21        this.taxajuros = A.taxajuros;
22    }
23
24    // Destrutor de classe
25    public void finalize() {}
26
27    // Calculo de exponeta
28    static double potencia(double n, double power) {
29        double res = 1;
30
31        for(double i = 1; i <= power; i++) {
32            res = res*n;
33        }
34        return res;
35    }
36
37    // Exibição dos elementos da tabela S.A.C/S.A.F/S.A.M/S.A.Alemão convertidos em Real e Decimal
38    public static void exibirTAB(int valor0, double valor1, double valor2, double valor3, double valor4) {
39        DecimalFormat decimalFormat = new DecimalFormat("R$#,##0.00");
40        decimalFormat.setRoundingMode(RoundingMode.DOWN);
41        System.out.println("Período: " + valor0 + "\tSaldo atual:\t" + decimalFormat.format(valor1)
42            + "\tAmortização:\t"+decimalFormat.format(valor2)+"\tJuros: "+decimalFormat.format(valor3)
43            + "\t\tPrestação:\t" + decimalFormat.format(valor4));
44    }
45
46    // Exibição os elementos do periodo 0 das tabelas S.A.C/S.A.F/S.A.M/S.A.Alemão convertidos em Real e Decimal
47    public static void exibir0(int valor0, double valor1, double valor2, double valor3, double valor4) {
48        DecimalFormat decimalFormat = new DecimalFormat("R$#,##0.00");
49        decimalFormat.setRoundingMode(RoundingMode.DOWN);
50        System.out.println("Período: " + valor0 + "\tSaldo atual:\t" + decimalFormat.format(valor1)
51            + "\tAmortização:\t"+decimalFormat.format(valor2)+"\t\tJuros:\t"+decimalFormat.format(valor3)
52            + "\t\tPrestação:\t" + decimalFormat.format(valor4));
53    }
54
55    // Exibição os elementos do periodo 0 do fundo de amortizacao da tabela S.A.A convertidos em Real e Decimal
56    public static void exibir1(int valor0, double valor1, double valor2, double valor3) {
57        DecimalFormat decimalFormat = new DecimalFormat("R$#,##0.00");
58        decimalFormat.setRoundingMode(RoundingMode.DOWN);

```

```

58     System.out.println("Período: " + valor0 + "\tDepósito:\t" + decimalFormat.format(valor1)
59     + "\tJuros:\t"+decimalFormat.format(valor2)+"\tMontante:\t"+decimalFormat.format(valor3));
60 }
61
62 // Exibição dos elementos da planilha financeira da tabela S.A.A convertidos em Real e Decimal
63 public static void exibirSAA1(int valor0, double valor1, double valor2, double valor3, double valor4) {
64     DecimalFormat decimalFormat = new DecimalFormat("R$#,##0.00");
65     decimalFormat.setRoundingMode(RoundingMode.DOWN);
66     System.out.println("Período: " + valor0 + "\tSaldo atual:\t" + decimalFormat.format(valor1)
67     + "\tAmortização:\t" + decimalFormat.format(valor2) + "\tJuros:\t" + decimalFormat.format(valor3)
68     + "\tPrestação:\t" + decimalFormat.format(valor4));
69 }
70
71 // Exibição os elementos do fundo de amortização da tabela S.A.A convertidos em Real e Decimal
72 public static void exibirSAA2(int valor0, double valor1, double valor2, double valor3) {
73     DecimalFormat decimalFormat = new DecimalFormat("R$#,##0.00");
74     decimalFormat.setRoundingMode(RoundingMode.DOWN);
75     System.out.println("Período: " + valor0 + "\tDepósito:\t" + decimalFormat.format(valor1) + "\tJuros:\t"
76     + decimalFormat.format(valor2)
77     + "\tMontante:\t" + decimalFormat.format(valor3));
78 }
79
80 // Método S.A.C - Sistema de Amortização Constante
81 public void SAC() {
82     DecimalFormat decimalFormat = new DecimalFormat("R$#,##0.00");
83     decimalFormat.setRoundingMode(RoundingMode.DOWN);
84     double amortizacao, juros, prestacao, tam=0, tj=0, tp=0;
85     amortizacao = saldoatual/periodo;

```

```

86     for (int i = 1; i<=periodo; i++) {
87         juros = saldoatual*taxajuros;
88         prestacao = amortizacao + juros;
89         saldoatual -= amortizacao;
90         tam = tam + amortizacao;
91         tj = tj + juros;
92         tp = tp + prestacao;
93         exibirTAB(i, saldoatual, amortizacao, juros, prestacao);
94     }
95     System.out.println("\nTotalizações - \tAmortização: "+decimalFormat.format(tam)+"\tJuros: "
96     +decimalFormat.format(tj)
97     +"\tPrestacao: "+decimalFormat.format(tp));
98 }
99
100 // Método S.A.F - Sistema de Amortização Franca - Tabela Price
101 public void SAF() {
102     DecimalFormat decimalFormat = new DecimalFormat("R$#,##0.00");
103     decimalFormat.setRoundingMode(RoundingMode.DOWN);
104     double coef = ((potencia((1+taxajuros), periodo))*taxajuros)/((potencia((1+taxajuros), periodo)-1));
105     double amortizacao=0, juros=0, saldodevedor, prestacao=0, tam=0, tj=0, tp=0;
106     int i = 0;
107     exibir0(i, saldoatual, amortizacao, juros, prestacao);
108     prestacao = saldoatual*coef;
109     for(i = 1; i<=periodo; i++) {
110         juros = saldoatual*taxajuros;
111         amortizacao = prestacao - juros;
112         saldodevedor = saldoatual + juros;
113         saldoatual = saldodevedor - prestacao;

```

```

114         tam += amortizacao;
115         tj += juros;
116         tp += prestacao;
117         exibirTAB(i, saldoatual, amortizacao, juros, prestacao);
118     }
119     System.out.println("Totalizações - \t\t\t\tAmortização:\t"+decimalFormat.format(tam)+"\tJuros: "
120     +decimalFormat.format(tj)
121     +"\t\t\tPrestacao:\t"+decimalFormat.format(tp));
122 }
123
124 // Subclasse de Amortização Americana para acomodar a captação média
125 public class AmortizacaoAmericana extends Amortizacao {
126     double captacaomedia;
127
128     // Construtor da subclasse
129     public AmortizacaoAmericana(double pr, double sa, double txj, double cm) {
130         super(pr, sa, txj);
131         captacaomedia = cm;
132     }
133
134     // Método S.A.A
135     public void SAA() {
136         DecimalFormat decimalFormat = new DecimalFormat("R$#,##0.00");
137         decimalFormat.setRoundingMode(RoundingMode.DOWN);
138         double amortizacao=0, jurosPF=0, prestacao=0, deposito, jurosFA=0, montante, tam=0, tjpf=0,
139         tp=0, tjfa=0, tm=0, td=0;
140         int i = 0;
141         exibir0(i, saldoatual, amortizacao, jurosPF, prestacao);

```

```

142     jurosPF = saldoatual * taxaJuros;
143     prestacao = jurosPF - amortizacao;
144     deposito = saldoatual*((captacaomedio)/(potencia((1+captacaomedio),periodo)-1));
145     montante = deposito;
146
147     System.out.println("\nPLANILHA FINANCEIRA\n");
148
149     for(i = 1; i<=periodo; i++) {
150         if(i==periodo) {
151             amortizacao = saldoatual;
152             saldoatual = 0;
153             prestacao = amortizacao + jurosPF;
154         }
155         tam += amortizacao;
156         tjpf += jurosPF;
157         tp += prestacao;
158         exibirSAA1(i, saldoatual, amortizacao, jurosPF, prestacao);
159     }
160
161     System.out.println("Totalizações\t - \tAmortização: "+decimalFormat.format(tam)+"\tJuros:\t"
162     +decimalFormat.format(tjpf)
163     +"\tPrestacao:\t"+decimalFormat.format(tp));
164
165     i=0;
166     System.out.println("\nFUNDO DE AMORTIZAÇÃO\n");
167     exibirI(i, deposito, jurosFA, montante);
168
169     for(i=1; i<=periodo; i++) {
170
171         if(i>1) {
172             jurosFA = montante * captacaomedio;
173         }
174         montante += jurosFA;
175         tjfa += jurosFA;
176         tm += montante;
177         td += deposito;
178         exibirSAA2(i, deposito, jurosFA, montante);
179     }
180
181     System.out.println("Totalizações - Depósito:\t"+ decimalFormat.format(td)
182     +"\tJuros:\t"+ decimalFormat.format(tjfa)+"\tMontante:\t"+ decimalFormat.format(tm));
183 }
184
185 // Método S.A.M - Sistema de Amortização Mista
186 public void SAM() {
187     DecimalFormat decimalFormat = new DecimalFormat("R$,##0.00");
188     decimalFormat.setRoundingMode(RoundingMode.DOWN);
189     double coef = ((potencia((1+taxajuros),periodo))*taxajuros)/((potencia((1+taxajuros),periodo)-1));
190     double amortizacao=0, amortizacaoSAC, juros=0, prestacaoSAF=0, prestacaoSAC, prestacaoSAM=0,
191     saldoatualSAC=saldoatual, tam=0, tj=0, tp=0;
192     int i = 0;
193     exibir0(i, saldoatual, amortizacao, juros, prestacaoSAM);
194     prestacaoSAF = saldoatual*coef;
195     amortizacaoSAC=(saldoatual/periodo);
196     for(i = 1; i<=periodo; i++) {
197         prestacaoSAC = amortizacaoSAC + (saldoatualSAC*taxajuros);
198         juros = saldoatual*taxajuros;
199         saldoatualSAC -= amortizacaoSAC;
200         prestacaoSAM = (prestacaoSAC + prestacaoSAF)/2;
201         amortizacao = prestacaoSAM - juros;
202         saldoatual -= amortizacao;
203         tam += amortizacao;
204         tj += juros;
205         tp += prestacaoSAM;
206         exibirTAB(i, saldoatual, amortizacao, juros, prestacaoSAM);
207     }
208     System.out.println("Totalizações - \t\t\t\t\tAmortização:\t"+decimalFormat.format(tam)
209     +"\tJuros:\t"+decimalFormat.format(tj)
210     +"\tPrestacao:\t"+ decimalFormat.format(tp));
211 }
212
213 // Método S.A.Alemão - Sistema de Amortização Alemão
214 public void SAALM() {
215     DecimalFormat decimalFormat = new DecimalFormat("R$,##0.00");
216     decimalFormat.setRoundingMode(RoundingMode.DOWN);
217     double amortizacaok=0, amortizacaol, juros, prestacao=0, tam=0, tj=0, tp=0;
218     int i = 0;
219     juros = taxajuros*saldoatual;
220     exibirTAB(i, saldoatual, amortizacaok, juros, prestacao);
221     prestacao = (saldoatual*taxajuros)/(1-potencia((1-taxajuros), periodo));
222     amortizacaol = prestacao * potencia((1-taxajuros),periodo-1);
223     amortizacaok = amortizacaol;
224     juros = prestacao - amortizacaok;
225     saldoatual -= amortizacaok;

```

```

226     i = 1;
227     exibirTAB(i, saldoatual, amortizacaok, juros, prestacao);
228     for(i = 2; i<=período; i++) {
229         amortizacaok = amortizacaok/(1-taxajuros);
230         juros = prestacao - amortizacaok;
231         saldoatual -= amortizacaok;
232         if (i == período) {
233             amortizacaok = 0;
234             juros = 0;
235         }
236         tam += amortizacaok;
237         tj += juros;
238         tp += prestacao;
239         exibirTAB(i, saldoatual, amortizacaok, juros, prestacao);
240     }
241     System.out.println("Totalizações - \t\t\t\t\tAmortização:\t"+decimalFormat.format(tam)
242     +"\tJuros:\t"+decimalFormat.format(tj)
243     +"\tPrestacao:\t"+ decimalFormat.format(tp));
244 }
245
246 // Função Main para testes
247 public static void main(String args[]) {
248     Locale.setDefault(Locale.US);
249     Scanner entrada = new Scanner(System.in);
250
251     System.out.println("Quantidade de parcelas: ");
252     double qtdMeses = entrada.nextDouble();
253

```

```

254     System.out.println("Valor a ser pago: ");
255     double valorPag = entrada.nextDouble();
256
257     System.out.println("Valor do juros (formato decimal): ");
258     double valorJuros = entrada.nextDouble();
259     while (valorJuros <= 0 || valorJuros > 1) {
260
261         valorJuros = entrada.nextDouble();
262
263     }
264
265     System.out.println("Escolha o número da amortização que será utilizada:\n"
266     + "(1) Sistema de Amortização Americano\n"
267     + "(2) Sistema de Amortização Francês\n"
268     + "(3) Sistema de Amortização Alemão\n"
269     + "(4) Sistema de Amortização Misto\n"
270     + "(5) Sistema de Amortização Constante\n");
271     int tipoAmort = entrada.nextInt();
272     while ( tipoAmort <= 0 || tipoAmort > 5) {
273         System.out.println("Valor inválido! Por favor, digite um valor correto.");
274         tipoAmort = entrada.nextInt();
275     }
276     Amortizacao calculo = new Amortizacao(qtdMeses,valorPag,valorJuros);
277
278     switch (tipoAmort) {
279
280     case 1:
281         System.out.println("Digite o valor da captura média: \n");

```

```

282         double capMed = entrada.nextDouble();
283         while (capMed < 0 || capMed > 1) {
284             System.out.println("Valor inválido! Por favor, digite um valor correto (entre 0 e 1).");
285             capMed = entrada.nextDouble();
286         }
287         AmortizacaoAmericana calc = calculo.new AmortizacaoAmericana(qtdMeses,valorPag,valorJuros, capMed);
288         calc.SAA();
289         break;
290
291     case 2:
292         calculo.SAF();
293         break;
294
295     case 3:
296         calculo.SAALM();
297         break;
298
299     case 4:
300         calculo.SAM();
301         break;
302
303     case 5:
304         calculo.SAC();
305         break;
306     }
307     System.out.println("Fim do programa.");
308     entrada.close();
309 }
310

```

4 TESTES DO PROGRAMA

Para verificar o funcionamento do programa, foram utilizados alguns exercícios sobre esse assunto em sites pela internet. Os exemplos usados nos testes serão apresentados a seguir.

4.1 Teste do Método S.A.C.

Um empréstimo de R\$ 60 000,00 deve ser devolvido de acordo com o sistema de amortizações constantes em 120 prestações mensais a taxa de juros de 1% ao mês. Construa a planilha referente as 5 primeiras prestações.

Resultado:

```

Quantidade de parcelas:
120
Valor a ser pago:
60000.00
Valor do juros (formato decimal):
0.01
Escolha o número da amortização que será utilizada:
(1) Sistema de Amortizacao Americano
(2) Sistema de Amortizacao Frances
(3) Sistema de Amortizacao Alemão
(4) Sistema de Amortizacao Misto
(5) Sistema de Amortizacao Constante
5

```

| | | | | | | | | |
|------------|--------------|--------------|--------------|-----------|--------|-----------|------------|-------------|
| Período: 1 | Saldo atual: | R\$59,500.00 | Amortização: | R\$500.00 | Juros: | R\$600.00 | Prestação: | R\$1,100.00 |
| Período: 2 | Saldo atual: | R\$59,000.00 | Amortização: | R\$500.00 | Juros: | R\$595.00 | Prestação: | R\$1,095.00 |
| Período: 3 | Saldo atual: | R\$58,500.00 | Amortização: | R\$500.00 | Juros: | R\$590.00 | Prestação: | R\$1,090.00 |
| Período: 4 | Saldo atual: | R\$58,000.00 | Amortização: | R\$500.00 | Juros: | R\$585.00 | Prestação: | R\$1,085.00 |
| Período: 5 | Saldo atual: | R\$57,500.00 | Amortização: | R\$500.00 | Juros: | R\$580.00 | Prestação: | R\$1,080.00 |

4.2 Teste do método S.A.F.

Uma geladeira é vendida, pelo pagamento à vista, por R\$ 1 200,00. Caso o consumidor queira parcelar o eletrodoméstico, será cobrada uma taxa de juros no valor de 2% ao mês. Considerando que uma pessoa comprou a geladeira em 12 prestações iguais, calcule o valor das parcelas utilizando o sistema Price.

Resultado:

```

Quantidade de parcelas:
12
Valor a ser pago:
1200
Valor do juros (formato decimal):
0.02
Escolha o número da amortização que será utilizada:
(1) Sistema de Amortizacao Americano
(2) Sistema de Amortizacao Frances
(3) Sistema de Amortizacao Alemao
(4) Sistema de Amortizacao Misto
(5) Sistema de Amortizacao Constante
2
Período: 0 Saldo atual: R$1,200.00 Amortização: R$0.00 Juros: R$0.00 Prestação: R$0.00
Período: 1 Saldo atual: R$1,110.52 Amortização: R$89.47 Juros: R$24.00 Prestação: R$113.47
Período: 2 Saldo atual: R$1,019.26 Amortização: R$91.26 Juros: R$22.21 Prestação: R$113.47
Período: 3 Saldo atual: R$926.18 Amortização: R$93.08 Juros: R$20.38 Prestação: R$113.47
Período: 4 Saldo atual: R$831.23 Amortização: R$94.94 Juros: R$18.52 Prestação: R$113.47
Período: 5 Saldo atual: R$734.38 Amortização: R$96.84 Juros: R$16.62 Prestação: R$113.47
Período: 6 Saldo atual: R$635.60 Amortização: R$98.78 Juros: R$14.68 Prestação: R$113.47
Período: 7 Saldo atual: R$534.84 Amortização: R$100.75 Juros: R$12.71 Prestação: R$113.47
Período: 8 Saldo atual: R$432.06 Amortização: R$102.77 Juros: R$10.69 Prestação: R$113.47
Período: 9 Saldo atual: R$327.23 Amortização: R$104.83 Juros: R$8.64 Prestação: R$113.47
Período: 10 Saldo atual: R$220.31 Amortização: R$106.92 Juros: R$6.54 Prestação: R$113.47
Período: 11 Saldo atual: R$111.24 Amortização: R$109.06 Juros: R$4.40 Prestação: R$113.47
Período: 12 Saldo atual: R$0.00 Amortização: R$111.24 Juros: R$2.22 Prestação: R$113.47
Totalizações - Amortização: R$1,199.99 Juros: R$161.65 Prestação: R$1,361.65
Fim do programa.

```

4.3 Teste do método de amortização americano

Construa a planilha e determine o valor total dos juros pagos pelo empréstimo referente a R\$ 25.250,00, pagos pelo sistema americano durante 5 meses, a uma taxa de 2,5% ao mês. **Resultado:**

```

Quantidade de parcelas:
5
Valor a ser pago:
25250
Valor do juros (formato decimal):
0.025
Escolha o número da amortização que será utilizada:
(1) Sistema de Amortizacao Americano
(2) Sistema de Amortizacao Frances
(3) Sistema de Amortizacao Alemao
(4) Sistema de Amortizacao Misto
(5) Sistema de Amortizacao Constante
1
Digite o valor da captacao media:
1
Período: 0 Saldo atual: R$25,250.00 Amortização: R$0.00 Juros: R$0.00 Prestação: R$0.00
PLANILHA FINANCEIRA
Período: 1 Saldo atual: R$25,250.00 Amortização: R$0.00 Juros: R$631.25 Prestação: R$631.25
Período: 2 Saldo atual: R$25,250.00 Amortização: R$0.00 Juros: R$631.25 Prestação: R$631.25
Período: 3 Saldo atual: R$25,250.00 Amortização: R$0.00 Juros: R$631.25 Prestação: R$631.25
Período: 4 Saldo atual: R$25,250.00 Amortização: R$0.00 Juros: R$631.25 Prestação: R$631.25
Período: 5 Saldo atual: R$0.00 Amortização: R$25,250.00 Juros: R$631.25 Prestação: R$25,881.25
Totalizações - Amortização: R$25,250.00 Juros: R$3,156.25 Prestação: R$28,406.25

```

4.4 Teste do método S.A.M.

Um cliente realizou um empréstimo de R\$10.000,00 e pretende pagá-lo após quatro meses, com uma taxa de juros de 1% a.m, utilizando o S.A.M. Construa a planilha. **Resultado:**

```

Quantidade de parcelas:
4
Valor a ser pago:
10000
Valor do juros (formato decimal):
0.01
Escolha o número da amortização que será utilizada:
(1) Sistema de Amortizacao Americano
(2) Sistema de Amortizacao Frances
(3) Sistema de Amortizacao Alemao
(4) Sistema de Amortizacao Misto
(5) Sistema de Amortizacao Constante
4
Período: 0      Saldo atual:   R$10,000.00   Amortização:  R$0.00       Juros:   R$0.00       Prestação:   R$0.00
Período: 1      Saldo atual:   R$7,518.59   Amortização:  R$2,481.40   Juros:   R$100.00      Prestação:   R$2,581.40
Período: 2      Saldo atual:   R$5,024.87   Amortização:  R$2,493.71   Juros:   R$75.18        Prestação:   R$2,568.90
Período: 3      Saldo atual:   R$2,518.71   Amortização:  R$2,506.15   Juros:   R$50.24        Prestação:   R$2,556.40
Período: 4      Saldo atual:   R$0.00       Amortização:  R$2,518.71   Juros:   R$25.18        Prestação:   R$2,543.90
Totalizações -  Saldo atual:   R$0.00       Amortização:  R$9,999.99   Juros:   R$250.62      Prestacao:   R$10,250.62
Fim do programa.

```

REFERÊNCIAS

ARIMA, Carlos Hideo. Economia e Finanças: Aula 6 – Recursos para o Projeto. **Fatec São Paulo**, 30 de set. de 2022. Acesso em: 30 de nov. de 2022.

AVELAR, Luiz. Sistema de Amortização Misto (SAM). In: SLIDES SHARE (ed.). Sistema de Amortização Misto (SAM). [S. l.], 1 abr. 2022. Disponível em: <<https://pt.slideshare.net/Snipermineiro/sistema-de-amortizacao-misto-sammfnun7edit2pdf>>. Acesso em: 01 dez. 2022.

BRASIL ESCOLA. **Exercícios sobre Financiamentos Utilizando a Tabela Price**. [S. l.], 2018. Disponível em: <<https://exercicios.brasilecola.uol.com.br/exercicios-matematica/exercicios-sobre-financiamentos-utilizando-tabela-price.htm>>. Acesso em: 01 dez. 2022.

BRASIL ESCOLA. **Exercícios sobre SAC**. [S. l.], 2018. Disponível em: <<https://exercicios.brasilecola.uol.com.br/exercicios-matematica/exercicios-sobre-sac-sistema-amortizacoes-constantess.htm#questao-1>>. Acesso em: 01 dez. 2022.

CARVALHO, Hélio Serio Leão. Sistema alemão de amortização. **YouTube**, 17 de out. de 2021. Disponível em: <<https://www.youtube.com/watch?v=eS9sTLEN7JU>>. Acesso em: 30 de nov. de 2022.

SILVA, Marcos Noé Pedro da. "Sistema Americano de Amortização "; **Brasil Escola**. Disponível em: <<https://brasilecola.uol.com.br/matematica/sistema-americano-amortizacao.htm>>. Acesso em 01 de dez. de 2022.

SISTEMAS de amortização: Conheça os principais e veja como calcular. **Capital Now**, 12 de nov. de 2019. Disponível em: <<https://capitalresearch.com.br/blog/sistemas-de-amortizacao/>>. Acesso em: 23 de nov. de 2022.

SODRÉ, Ulysses. Sistema alemão de Amortização. **Universidade Estadual de Londrina**, 29 de jul. de 2020. Disponível em: <<http://www.uel.br/projetos/matessencial/basico/financeira/alemao.html>>. Acesso em: 26 de nov. de 2022.

SODRÉ, Ulysses. Sistemas de Amortização. **Universidade Estadual de Londrina**, 29 de jul. de 2020. Disponível em: <<http://www.uel.br/projetos/matessencial/basico/financeira/amortiza.html>>. Acesso em: 23 de nov. de 2022.