

MAC 115 – Introdução à Ciência da Computação

Segundo Exercício-Programa

De olho no bolso

1 Introdução

Uma situação com a qual nos deparamos freqüentemente quando vamos adquirir um bem de consumo é saber qual é a taxa de juros¹ embutida em um financiamento. Em geral, sabemos somente o valor à vista do bem, V_{Bem} e, caso optemos por um crediário, o número de prestações n e o valor de cada prestação mensal P_k , $1 \leq k \leq n$. A taxa mensal de juros $j \geq 0$ não é revelada na maioria das vezes.

Vamos supor que todas as prestações possuem um mesmo valor P e que a primeira das prestações é paga à vista, ou seja, a entrada do financiamento. Trazendo todas as prestações vincendas para o presente, podemos² escrever:

$$V_{Bem} = P + \frac{P}{1+j} + \frac{P}{(1+j)^2} + \dots + \frac{P}{(1+j)^{n-1}} = PS$$

onde definimos S pela somatória

$$S = 1 + \frac{1}{1+j} + \frac{1}{(1+j)^2} + \dots + \frac{1}{(1+j)^{n-1}}.$$

Observe ainda que podemos³ simplificar o cálculo de S por

$$S = \begin{cases} \frac{(1-(\frac{1}{1+j})^n)}{1-\frac{1}{1+j}} & \text{se } j > 0; \\ n & \text{se } j = 0. \end{cases}$$

Caso conheçamos o valores de V_{Bem} , n e P , e estejamos interessados em saber que taxa de juros j está sendo praticada, precisamos resolver um polinômio de grau $n-1$ em j que, todavia, não possui um método genérico para a obtenção de uma solução exata. Usualmente satisfazemo-nos com uma solução aproximada de uma tal raiz, obtida através de métodos computacionais. A determinação da taxa mensal de juros j pode então ser feita pela determinação da⁴ raiz real positiva da função:

$$f(j) = S - \frac{V_{Bem}}{P}.$$

Para a determinação de tal raiz, utilizaremos neste EP um método conhecido por método de Muller, a ser descrito a posteriori.

Definimos o *erro relativo* de x em relação a y como sendo:

$$\text{ErroRel}(x, y) = \begin{cases} |\frac{x-y}{y}|, & \text{se } y \neq 0; \\ 1, & \text{se } y = 0. \end{cases}$$

¹Para facilitar a nossa vida, as taxas de juros serão números reais e não porcentagens. Assim, 5% é representado por 0.05.

²Verifique!

³Por que podemos?

⁴Por que esta raiz positiva é única?

2 O que fazer?

OBS: Neste EP, todos os números reais deverão ser impressos com o formato `%.20g` para melhor precisão nas impressões, embora a precisão de um `double` seja menor que vinte casas. Devem ser entregues o código fonte das fases abaixo e a saída da execução dos programas com as entradas de dados vistas nos exemplos.

- A) Programa preliminar [3 pontos]: escreva um programa que leia uma sequência de “triplos” e, para cada tripla (V_{Bem}, n, j) , computa o valor da prestação P correspondente a um financiamento de um bem de valor $V_{Bem} > 0$ em $n \geq 1$ prestações a juros mensais $j \geq 0$. A sequência é terminada por um número negativo.
- B) Programa principal [7 pontos]: faça um programa que leia uma sequência de quádruplas $(V_{Bem}, n, P, \epsilon)$ terminada por um número negativo e que calcula, para cada quádrupla, a taxa de juros j com precisão ϵ correspondente ao financiamento de um bem de valor $V_{Bem} > 0$ com $n \geq 1$ prestações de valor $P > 0$.

Parta das aproximações iniciais⁵ 0 , $0.3 \times \frac{P}{V_{Bem}-P}$ e $\frac{P}{(V_{Bem}-P)}$ e imprima todas as aproximações obtidas e respectivos erros relativos.

OBS: Não é permitido o uso de nenhuma função da biblioteca `math.h` neste EP. Você deve implementar as funções de protótipo abaixo especificadas que computam respectivamente os valores de: `x` elevado a `n`; a raiz quadrada positiva de x com precisão ϵ ; o valor absoluto de `x`; a taxa de juros com precisão `epsilon` correspondente ao financiamento de um bem de valor `vbem` com `n` prestações de valor `prestacao`; o erro relativo de `x` em relação a `y`.

```
double potencia(double x,int n);
double raiz_quadrada(double x, double epsilon);
double valor_absoluto(double x);
double juros(double vbem, int n,
             double prestacao, double epsilon);
double erro_relativo(double x, double y);
```

- C) Programa Bônus [1 ponto]: faça um programa que faça tudo o que é feito na parte B) exceto as aproximações intermediárias de j e, além disso para cada quádrupla, calcule a aproximação dos juros j' obtida pela fórmula:

$$j' = \frac{1 - \frac{V_{Bem}}{nP}}{\frac{n-1}{2}}$$

e calcule o erro relativo desta aproximação j' em relação à aproximação final de j obtida na parte B).

3 Exemplos

Parte A):

⁵Para os momentos de tédio: prove que a taxa de juros está no intervalo $[0, \frac{P}{V_{Bem}-P}]$. Verifique que os juros podem ser aproximados por $\frac{1 - \frac{V_{Bem}}{nP}}{\frac{n-1}{2}}$ quando o produto nj for suficientemente pequeno. Por quê?

Digite respectivamente os valores de

Valor NPrest Juros

1000 10 0

Prestacao = 100

Valor NPrest Juros

1000 10 0.001

Prestacao = 100.45037421217611495

Valor NPrest Juros

1000 10 0.01

Prestacao = 104.53670945660525149

Valor NPrest Juros

1000 10 0.10

Prestacao = 147.95035898410148434

Valor NPrest Juros

1000 100 0

Prestacao = 10

Valor NPrest Juros

1000 100 0.001

Prestacao = 10.502824125921280896

Valor NPrest Juros

1000 100 0.01

Prestacao = 15.708656559792597562

Valor NPrest Juros

1000 100 0.10

Prestacao = 90.915688271097010897

Valor NPrest Juros

-1

Acabou!

Parte B):

Digite respectivamente os valores de

Valor NPrest Prestacao ErroRelativo

1000 10 100 0.0000000001

Juros = 0

Valor NPrest Prestacao ErroRelativo

1000 10 100.45037421217615758 0.0000000001

j=0.0056536030318869359007 erro=18.751547182647396284

j=0.0080813183969970468767 erro=0.30041080500085609595

j=0.00071532691846834506842 erro=10.297377728075341352

j=0.00065365110898049779645 erro=0.094355855349260053799

j=0.00099468630971197985469 erro=0.34285703683830914823

j=0.00099416945537046772126 erro=0.00051988555745713653684

j=0.0010000180542310795669 erro=0.0058484932705628713337

j=0.0010000188918076579735 erro=8.3756075536991461687e-07

j=0.0010000000053131999876 erro=1.888649435763819454e-05

j=0.0010000000054336960487 erro=1.204960603909550236e-10

j=0.00099999999964784118053 erro=5.7858548701664206877e-09

```

j=0.00099999999964389772039 erro=3.9434601431685515265e-12
Juros = 0.00099999999964389772039
Valor NPrest Prestacao ErroRelativo
1000 10 104.53670945660525149 0.0000000001
j=0.043780186782295719972 erro=1.666511301794849631
j=0.038296890344673199802 erro=0.14317863378129874752
j=0.013126578240249502441 erro=1.9175074908131788387
j=0.012816473981137733659 erro=0.02419575458649200761
j=0.010065875617689911542 erro=0.27325972105336587825
j=0.010062394398205835172 erro=0.00034596333102358670965
j=0.010000159356518238327 erro=0.0062234049947445298917
j=0.010000155030051379942 erro=4.3263997861861115152e-07
j=0.010000000021579157072 erro=1.5500847188837485577e-05
j=0.010000000021282196699 erro=2.9696037232656882125e-11
Juros = 0.010000000021282196699
Valor NPrest Prestacao ErroRelativo
1000 10 147.95035898410148434 0.0000000001
j=0.15987941241611053345 erro=0.086071911638113549059
j=0.093874742906318822677 erro=0.70311425061009524828
j=0.094639384915063740067 erro=0.0080795327382057972182
j=0.10008236450933291617 erro=0.054385002002641590069
j=0.10007698924394124729 erro=5.3711301991374657419e-05
j=0.099999847534489774703 erro=0.00077141827066152818954
j=0.099999852678037459697 erro=5.1435552625802560297e-08
j=0.10000000001960264451 erro=1.4734156515592804707e-06
j=0.10000000001926631632 erro=3.3632818745758530992e-12
Juros = 0.10000000001926631632
Valor NPrest Prestacao ErroRelativo
1000 100 10 0.0000000001
Juros = 0
Valor NPrest Prestacao ErroRelativo
1000 100 10.502824125921280896 0.0000000001
j=0.0042113920784191127833 erro=1.5203790337084783779
j=0.0035292256714529977868 erro=0.19329067349928463759
j=0.0014254203989245109886 erro=1.4759191562824705013
j=0.0013685518609166101058 erro=0.041553805618891374862
j=0.0010200411345032776161 erro=0.34166340417540547181
j=0.0010185771907829876091 erro=0.0014372437685990816728
j=0.0010001608343940847964 erro=0.0184133948816939709
j=0.0010001547298470238063 erro=6.1036026514855310139e-06
j=0.0010000001033070194268 erro=0.00015462652403037415415
j=0.001000000101308496335 erro=1.9985228893416859235e-09
j=0.0010000000000069085571 erro=1.0130158777723567231e-07
j=0.0010000000000068424207 erro=6.6136332521163241994e-14
Juros = 0.0010000000000068424207
Valor NPrest Prestacao ErroRelativo
1000 100 15.708656559792595786 0.0000000001
j=0.014887886479618646363 erro=0.071969257259555774464

```

```

j=0.0094864671488875795119 erro=0.56938154593878065324
j=0.0095456056705154404551 erro=0.0061953660845773866547
j=0.01000722567207164293 erro=0.046128669092024218723
j=0.010006791482794107767 erro=4.3389459876496618734e-05
j=0.0099999872286009681283 erro=0.00068042628831352781719
j=0.0099999876242435441209 erro=3.9564306563089418896e-08
j=0.010000000001436640945 erro=1.2377193095046011142e-06
j=0.010000000001414039233 erro=2.2601712165305849009e-12
Juros = 0.010000000001414039233
Valor NPrest Prestacao ErroRelativo
1000 100 90.915688271097010897 0.0000000001
j=0.1000077257826280791 erro=2.5706347158275649948e-06
j=0.10000000104311776439 erro=7.7247394297365863203e-05
j=0.10000000102604499064 erro=1.7072773575118585516e-10
j=0.10000000000000033501 erro=1.026041640539397954e-08
j=0.100000000000000332234 erro=2.7755575615627991529e-16
Juros = 0.100000000000000332234
Valor NPrest Prestacao ErroRelativo
-1
Acabou!

```

Parte C):

```

Digite respectivamente os valores de
Valor NPrest Prestacao ErroRelativo
1000 10 100 0.0000000001
j = 0
j' = 0
erro = 1
Valor NPrest Prestacao ErroRelativo
1000 10 100.45037421217615758 0.0000000001
j = 0.000999999999964389772039
j' = 0.0011208873429004238657
erro = 0.12088734329957440128
Valor NPrest Prestacao ErroRelativo
1000 10 104.53670945660525149 0.0000000001
j = 0.010000000021282196699
j' = 0.010849560599782670858
erro = 0.084956057669242260322
Valor NPrest Prestacao ErroRelativo
1000 10 147.95035898410148434 0.0000000001
j = 0.10000000001926631632
j' = 0.081024404593121254758
erro = 0.18975595422489163755
Valor NPrest Prestacao ErroRelativo
1000 100 10 0.0000000001
j = 0
j' = 0

```

```

erro = 1
Valor NPrest Prestacao ErroRelativo
1000 100 10.502824125921280896 0.0000000001
j = 0.00100000000000068424207
j' = 0.000977043577280335869
erro = 0.022956422726349474256
Valor NPrest Prestacao ErroRelativo
1000 100 15.708656559792595786 0.0000000001
j = 0.0100000000001414039233
j' = 0.0074164964173961348687
erro = 0.25835035836525865882
Valor NPrest Prestacao ErroRelativo
1000 100 90.915688271097010897 0.0000000001
j = 0.100000000000000332234
j' = 0.018163428208749984871
erro = 0.81836571791250622976
Valor NPrest Prestacao ErroRelativo
-1
Acabou!

```

4 O método de Muller

1. Sejam x_{n-2} , x_{n-1} e x_n , três aproximações distintas de uma raiz de f e seja ϵ a precisão desejada. Se f é anulada nalgum dos pontos, já encontramos a raiz. Caso contrário, procuramos uma nova aproximação.
2. Para índices apropriados i , sejam $f_i = f(x_i)$, $h_i = x_i - x_{i-1}$ e $q_i = \frac{h_i}{h_{i-1}}$.
3. Seja $A = q_n f_n - q_n(1 + q_n)f_{n-1} + q_n^2 f_{n-2}$.
4. Seja $B = (2q_n + 1)f_n - (1 + q_n)^2 f_{n-1} - q_n^2 f_{n-2}$.
5. Seja $C = (1 + q_n)f_n$.
6. Definindo $q' = -\frac{2C}{B + \sqrt{B^2 - 4AC}}$,
7. $q'' = -\frac{2C}{B - \sqrt{B^2 - 4AC}}$ e
8. Seja q_{n+1} o elemento de $\{q', q''\}$ com menor valor absoluto.
9. A nova aproximação para a raiz de f é obtida por $x_{n+1} = x_n + h_n q_{n+1}$.

Repetimos o processo acima com as três últimas aproximações calculadas (x_{n-1} , x_n , x_{n+1}) até que a precisão desejada seja atingida, ou seja, até que

$$\text{ErroRel}(x_n, x_{n+1}) < \epsilon \text{ ou } f(x_{n+1}) = 0.$$