Cálculo Numérico **Nome** _____
Prof. Luis Peñaranda DRE _____
Prova 1 – 2015.1
28/05/2015
Tempo máximo: 120 minutos

Os pontos correspondentes a cada item estão especificados à esquerda do texto. O total de pontos de cada exercício está especificado depois do enunciado de cada exercício, à direita. O máximo de pontos possível na prova é 10.

1. Usar a norma IEEE-754 para responder as questões seguintes.

[1] (a) Representar o número 0.51 em virgula flutuante com dupla precisão, indicando claramente quais bits formam o sinal, o exponente e a mantissa. Essa representação, é exata?

[1] (b) Modifique a solução do item anterior para representar o número 2.04 com precisão simples (este item será zerado se não usar a solução do item anterior).

Total questão 1: 2 pontos

Gabarito: Vamos fazer o algoritmo de multiplicação sucessiva por 2, e vamos a usar como bits do número o primeiro dígito significativo dos produtos.

$$
\begin{aligned}
0.51 \times 2 &= 1.02 \\
0.02 \times 2 &= 0.04 \\
0.04 \times 2 &= 0.08 \\
0.08 \times 2 &= 0.16 \\
0.16 \times 2 &= 0.32 \\
0.32 \times 2 &= 0.64 \\
0.64 \times 2 &= 1.28 \\
0.28 \times 2 &= 0.56 \\
0.56 \times 2 &= 1.12 \\
0.12 \times 2 &= 0.24 \\
0.24 \times 2 &= 0.48 \\
0.48 \times 2 &= 0.96
\end{aligned}
\qquad
\begin{aligned}
0.96 \times 2 &= 1.92 \\
0.92 \times 2 &= 1.84 \\
0.84 \times 2 &= 1.68 \\
0.68 \times 2 &= 1.36 \\
0.36 \times 2 &= 0.72 \\
0.72 \times 2 &= 1.44 \\
0.44 \times 2 &= 0.88 \\
0.88 \times 2 &= 1.76 \\
0.76 \times 2 &= 1.52 \\
0.52 \times 2 &= 1.04 \\
0.04 \times 2 &= \ldots
\end{aligned}
$$

Aqui notamos que, no próximo passo, deveríamos multiplicar 0.04 por dois. Porém, 0.04 já apareceu no processo, na terceira linha. Isso quer dizer que repetiríamos o processo indefinidamente. Ou seja, o número 0.51 em binário é periódico. A representação dele seria $0.1\overline{00000101000111101011}$, repetindo as últimas vinte cifras. Para representar o número em dupla precisão, precisamos os primeiros 53

dígitos do número (vamos escrever espaços entre os blocos do número que repetem-se: 0.10 00001010001111010111 00001010001111010111 00001010001. Devemos obter um número da forma 1...., para o qual precisamos de multiplicar por 2^1 o número anterior. Ou seja, o exponente do número na IEEE-754 seria -1. Para calcular a representação do exponente, considerando que temos onze bits para representar um inteiro sem sinal, adicionamos $2^{10} - 1$ ao exponente calculado e representamos como inteiro sem sinal. Ficaria então 01111111110. O sinal é +, ou seja, um bit 0.

A representação do número decimal 0.51 em um registro de precisão dupla ficaria:

- sinal: 0
- exponente: 01111111110
- mantissa (sem o primer bit 1): 0000010100011110101110000101000111101011100001010001

O número 2.04 é o número anterior multiplicado por 2^2. Então, vamos usar a mesma mantissa (embora troncada em 23 bits) e vamos adicionar 2 ao exponente, ou seja, $-1+2=1$. O novo exponente deve ser representado em 8 bits, ou seja, adicionando 2^7-1. O exponente ficaria então $(2^7-1)+1=2^7$ e seria representado por 10000000.

A representação do número decimal 2.04 em um registro de precisão simples ficaria:

- sinal: 0
- exponente: 10000000.
- mantissa (sem o primer bit 1): 00000101000111101011100

2. Considere a função polinomial $f(x) = 3x^3 + x - 1$.

$1\frac{1}{2}$ (a) Use a regra dos sinais de Descartes para saber quantas raízes positivas tem f.

1 (b) Se a resposta do item anterior for afirmativa, calcule intervalos de isolação para *todas* as raízes positivas de f, usando qualquer método.

$1\frac{1}{2}$ (c) Escolha um dos intervalos de isolação calculado no item anterior. Quantas iterações do refinamento por bisecção serão necessários para obter um intervalo de isolação de comprimento 10^{-6}?

Total questão 2: 4 pontos

Gabarito: A regra dos sinais de Descartes estabelece que a diferencia entre o número m de mudanças de sinal dos coeficientes de um polinômio e a quantidade de raízes

reais positivas r^+ é par, sendo $m \geq r^+$. Como em nosso caso $m = 1$, podemos afirmar que $r^+ = 1$. Ou seja, f tem exatamente uma raíz real positiva.

Como sabemos que a raíz é positiva, vamos avaliar f nos pontos arbitrários 0 e 10. Se eles tiverem diferentes sinais, podemos concluir que, no intervalo $(0, 10)$, f tem exatamente una raíz.

$$f(0) = 3 \times 0^3 + 0 - 1 = -1$$

$$f(10) = 3 \times 10^3 + 10 - 1 = 1009$$

Então, f tem exatamente uma raíz real positiva no intervalo $(0, 10)$ (notar que o intervalo é aberto, pois nem 0 nem 10 são raízes).

Temos um intervalo de isolação de comprimento $10 - 0 = 10$ (o comprimento pode mudar em função do intervalo escolhido no item anterior). Devemos atingir um intervalo de comprimento 10^{-6} dividindo o comprimento por 2 em cada passo da biseção. Ou seja, fazendo p passos, dividimos o comprimento do intervalo em 2^p. Queremos então que $\frac{10}{2^p} = 10^{-6}$. Ou seja, $2^p = 10^7$. Então $p = \log_2(10^7)$.

Nota: todas as raízes (complexas) de f podem ser calculadas em *Scilab*, usando os comandos seguintes.

```
v=[3 0 1 -1];
r=roots(v);
```

3. Dado o sistema seguinte:

$$\begin{cases} 4\ x + 6\,y - 3\,z = 11 \\ -3\,x + 3\,y - \ z = \ 2 \\ 4\ x - 4\,y - 4\,z = -2\,. \end{cases}$$

$1\frac{1}{2}$ (a) Resolva usando fatoração LU.

1 (b) Utilize o critério das linhas para saber se o método de Gauss-Jacobi converge à solução para qualquer valor da aproximação inicial x_0.

$1\frac{1}{2}$ (c) Aproxime a solução usando o método de Gauss-Jacobi, com $x_0 = (0, 1, 0)^T$. Faça só duas iterações do método. O método converge?

Total questão 3: 4 pontos

Gabarito:

As forma matricial do sistema é $A\overline{x} = b$, onde: $A = \begin{bmatrix} 4 & 6 & -3 \\ -3 & 3 & -1 \\ 4 & -4 & -4 \end{bmatrix}$, $\overline{x} = (x_0, x_1, x_2)^T$ e $b = (11, 2, -2)^T$.

A fatoração LU da matriz A é a seguinte. $PA = LU$, onde $P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$, $L = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ -\frac{3}{4} & -\frac{3}{4} & 1 \end{bmatrix}$, e $U = \begin{bmatrix} 4 & 6 & -3 \\ 0 & -10 & -1 \\ 0 & 0 & -4 \end{bmatrix}$.

Para resolver o sistema usando essa fatoração, fazemos dois passos.

(i) $Ly = Pb$, calculando $Pb = (11, -2, 2)^T$. O sistema fica:

$$\begin{cases} y_0 = 11 \\ y_0 + y_1 = -2 \\ -\frac{3}{4}\, y_0 - \frac{3}{4}\, y_1 + y_2 = 2 \end{cases}.$$

A solução do sistema é $y = (11, -13, \frac{1}{2})^T$.

(ii) Finalmente, $Ux = y$. O sistema é:

$$\begin{cases} 4\, x_0 + 6\, x_1 - 3\, x_2 = 11 \\ -10\, x_1 - x_2 = -13 \\ -4\, x_2 = \frac{1}{2} \end{cases}.$$

A solução do sistema é $x = (\frac{11}{16}, \frac{21}{16}, -\frac{1}{8})^T$.

O critério das linhas estabelece que, para cada linha da matriz, se o valor absoluto do elemento na diagonal for maior do que a soma dos restantes elementos da linha, então o método de Gauss-Jacobi converge para qualquer aproximação inicial. Esse critério não é satisfeito pela matriz A, então não temos a garantia de escolher qualquer aproximação inicial. (Porém, isso não quer dizer que o método não converge nunca.)

Vamos calcular a matriz C e o vetor g, para converter o sistema $Ax = b$ em um sistema equivalente $x = Cx + g$:

$C = \begin{bmatrix} 0 & -\frac{3}{2} & \frac{3}{4} \\ 1 & 0 & \frac{1}{3} \\ 1 & -1 & 0 \end{bmatrix}$ e $g = (\frac{11}{4}, \frac{2}{3}, \frac{1}{2})^T$.

Primeira iteração

$$x_1^{(1)} = -\frac{3}{2} + \frac{11}{4} = \frac{5}{4}$$
$$x_2^{(1)} = \frac{2}{3}$$
$$x_3^{(1)} = -1 + \frac{1}{2} = -\frac{1}{2}$$

Então, $x^{(1)} = (\frac{5}{4}, \frac{2}{3}, -\frac{1}{2})^T$.

Segunda iteração

$$x_1^{(2)} = -\frac{3}{2}\frac{2}{3} - \frac{3}{4}\frac{1}{2} + \frac{11}{4} = \frac{11}{8}$$
$$x_2^{(2)} = \frac{5}{4} - \frac{1}{3}\frac{1}{2} + \frac{1}{2} = \frac{19}{12}$$
$$x_3^{(2)} = \frac{5}{4} - \frac{2}{3} + \frac{1}{2} = \frac{13}{12}$$

Então, $x^{(2)} = (\frac{11}{8}, \frac{19}{12}, \frac{13}{12})^T$.

Intuitivamente, observamos que o método converge, pois $\|x^{(1)} - x^{(0)}\| = \sqrt{1,9235} > \|x^{(2)} - x^{(1)}\| = \sqrt{1,1959}$. Porém, isso não prova que o método converge.

Nota: a solução pode ser facilmente achada em *Scilab*, usando os comandos seguintes.

```
A=[4,6,-3;-3,3,-1;4,-4,-4]
b=[-11;-2;2]
linsolve(A,b)
```