LISTA 3 Interpolação

(As questões sinalizadas com **(**)** *deverão ser entregues até o dia* 19 *de Outubro,* 23 : 59 hrs*)*

1. Sejam $p1, p2, p3, p4, p5$ pontos da parábola $y = 3x^2 + 2x^2 - 4x + 7$. Quantos polinômios de grau 4 passam por esses 5 pontos?. Justifique

2. **(**)** Seja $P(x)$ um polinômio de grau n que passa pelos pontos $(i, -i)$ $(i = 1, \dots, n)$ e tal que tem termo independente igual a $(-1)^n$. Demonstre que para todo número natural $k > n$, tem-se que $P(k) = \binom{k-1}{n} - k$.

3. Considere a função $f(x) = \cos x$, para $x \in [0\ \pi]$. Determine o minimo de pontos a considerar no intervalo dado para que o erro máximo da aproximação de $f(x)$ por um polinômio interpolador nesses pontos seja inferior a 0.5

4. Pode um polinômio de grau 4 interceptar um polinômio de grau 5 em 6 pontos?. Justifique.

5. Considere o polinômio $p_9(x)$ que interpola $g(x) = e^{-2x}$ em 10 pontos $x_i = \frac{i}{9}$ $(i = 0, \dots 9)$.
   (a) Determine um limite superior para $\left|f(\frac{1}{2}) - p_9(\frac{1}{2})\right|$
   (b) Quantos dígitos significativos corretos podem ser garantidos se $p_9(\frac{1}{2})$ é usado para aproximar e ?

6. Use diferenças divididas de Newton para determinar quantos polinômios de grau d $(0 \leq d \leq 5)$ passam pelos pontos $(-1, -5)$, $(0, -1)$, $(2, 1)$, $(3, 11)$,

7. Escreva um programa em MATLAB que permita construir o polinômio interpolador de n pontos dados usando o método de Newton, e de tal forma que se são adicionados novos pontos o programa seja capáz de reutilizar o já calculado.

8. **(**)** Seja $f(x) = \sin(cx + d)$ $(c, d \in \mathbb{R})$. Considere a sequência $\{x_1, x_2, \cdots\}$, tal que $\exists n_0$ de modo que $\forall n > n_0$, todos os elementos da sequência são diferentes (i.e, $x_{n_0} \neq x_{n_0+1} \neq x_{n_0+2} \neq \cdots$). Demonstre que se $\Delta[x_i] = f(x_i)$, então a $k-$ésima diferença dividida de Newton satisfaz:

$$\lim_{k \to \infty} \Delta[x_1, x_2, \cdots, x_k] = 0$$

9. Considere a seguinte tabela de valores de uma função f

x_i	-1.25	3.75	6.25	1.25
$f(x_i)$	0.25328	61.547	959.42	3.9482

   (a) Construa o polinômio interpolador de grau dois que lhe permite obter a melhor aproximação para o valor da função no ponto $x = 0$.
   (b) Calcule uma aproximação para $f(0)$, utilizando o polinômio interpolador obtido no item anterior. Indique uma estimativa para o erro absoluto que se comete nessa aproximação.

10. Nesta questão é estudado o fenómeno de Runge, o qual pode acontecer na interpolação numérica: Considere

$$f(x) = \frac{1}{1 + 16x^2}, \quad x \in [-1\ 1]$$

Usando interpolação de Lagrange ache e faça o plot do polinômio interpolador em n pontos. Mostre todas as interpolações separadamente (total de 6 plots), mas use a mesma janela: $x \in [-1\ 1]$ e $y \in [-0.5\ \ 1.5]$. Discuta os resultados. Considere $y_i = f(x_i)$, $i = 1, ..., n$. Onde os valores x_i são dados debaixo

   (a) Pontos equidistantes $x_i = -1 + 2\frac{(i-1)}{n-1}$, $n = 10;\ 20;\ 40$.
   (b) Pontos de Chebyshev $x_i = \cos(\frac{(2i-1)\pi}{2n})$ $n = 10; 20; 40$.

11. Suponha quer-se construir a parte superior do cachorro da Figura 1. Para isto três splines cúbicos ancorados (também chamado de spline cúbico fixado) serão construidos, utilizando os dados da tabela a seguir:

curva 1			curva 2			curva 3		
x	$f(x)$	$f'(x)$	x	$f(x)$	$f'(x)$	x	$f(x)$	$f'(x)$
1	3	1	17	4.5	3	27.7	4.1	0.33
2	3.7		20	7.0		28	4.3	
5	3.9		23	6.1		29	4.1	
6	4.2		24	5.6		30	3	-1.5
7	5.7		25	5.8				
8	6.6		27	5.2				
10	7.1		27.7	4.1	-4			
13	6.7							
17	4.5	-0.67						

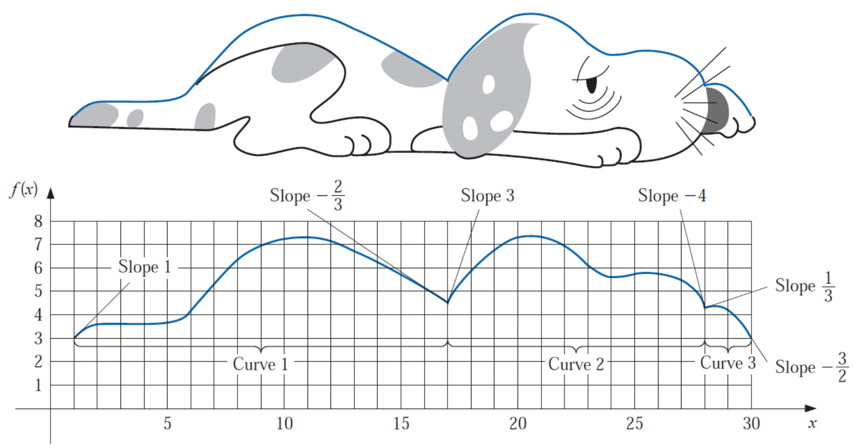

Figura 1

(a) Escreva os sistemas de equações que tem que ser resolvido para construir esses splines ancorados.

(b) Use o item anterior para construir no computador uma curva que represente a parte superior do cachoro.

12. **(**)** Nesta questão o objetivo é construir e representar graficamente *splines cúbicos paramétricos*. O problema consiste em, dados $n+2$ pontos no plano: (x_1, y_1), $(\hat{x}_1, \hat{y}_1)$, $(x_2, y_2), \cdots, (x_n, y_n)$, $(\hat{x}_n, \hat{y}_n)$, construir uma curva com curvatura suave que interpole os pontos (x_i, y_i) $(i = 1, \ldots, n)$ e de tal forma que a curva sai de (x_1, y_1) tangente ao vetor $(\hat{x}_1 - x_1, \hat{y}_1 - y_1)$ e entra/atinge ao ponto (x_n, y_n) tangencialmente ao vetor $(\hat{x}_n - x_n, \hat{y}_n - y_n)$ ($(\hat{x}_1, \hat{y}_1)$ e $(\hat{x}_n, \hat{y}_n)$ são chamados de pontos guias). Note que quando $x_1 < x_2 < .... < x_n$ essa curva coincide com um spline cúbico ancorado (tipo os splines da questão 11 acima). Mas, no caso dos x_i não serem estritamente crescentes, splines cúbicos ancorados obviamente não podem ser usados. Note também que para o caso particular $n = 2$, a curva vai coincidir com uma curva de Bézier. Mas, para $n > 2$ não. Por exemplo, na figura 2, para o spline paramétrico que sai de A e termina em E, tem-se que $n = 5$ e os pontos (x_i, y_i) são A, B, C, D, E, nessa ordem. Os pontos guias são A$'$=$(\hat{x}_1, \hat{y}_1)$ e E$'$=$(\hat{x}_5, \hat{y}_5)$.

(a) Construia matemáticamente o *spline cúbico paramétrico* $(x(t), y(t))$ (Para isto considere $t \in [0\ 1]$, selecione n pontos diferentes nesse intervalo, e aplique o aprendido nas aulas de interpolação spline para determinar $x(t)$ e $y(t)$)

(b) Implemente em MATLAB um programa que permita construir *splines cúbicos paramétricos* em um modo gráfico iterativo: O programa deve permitir marcar os pontos com o mouse na ordem: (x_1, y_1), $(\hat{x}_1, \hat{y}_1)$, $(x_2, y_2), \cdots, (x_n, y_n)$, $(\hat{x}_n, \hat{y}_n)$. E deve permitir fazer isto quantas vezes seja necessário para representar a curva desejada (Ou seja, se para representar a curva são necessários k *splines cúbicos paramétricos*, o programa deve permitir ir constuindo cada um deles, sem interrupção). Teste seu programa gerando as curvas das Figuras 2, 3 e 4.

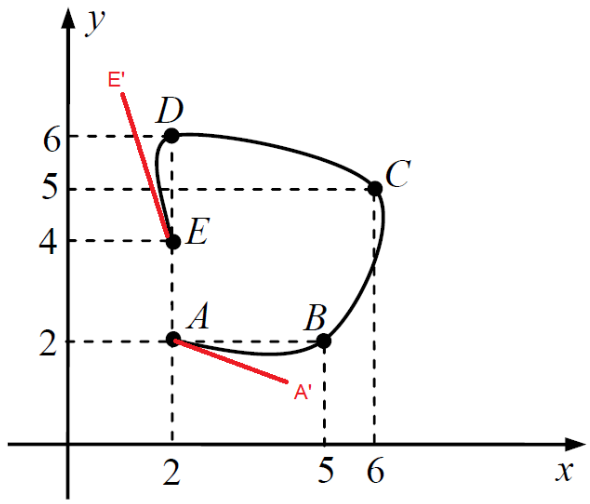

Figura 2

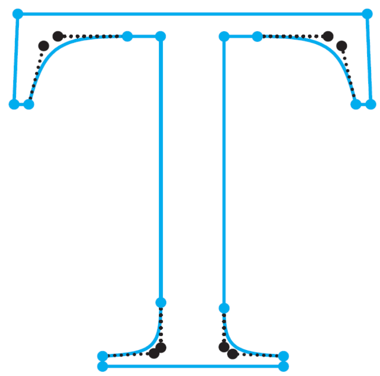

Figura 3

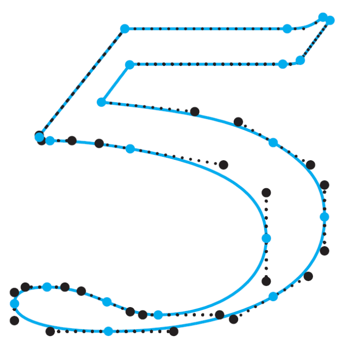

Figura 4