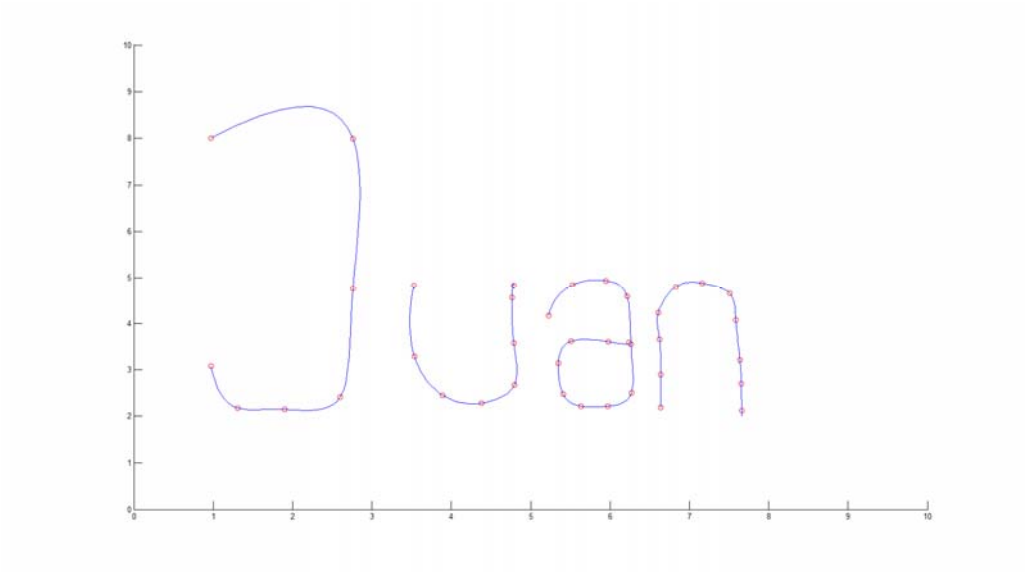
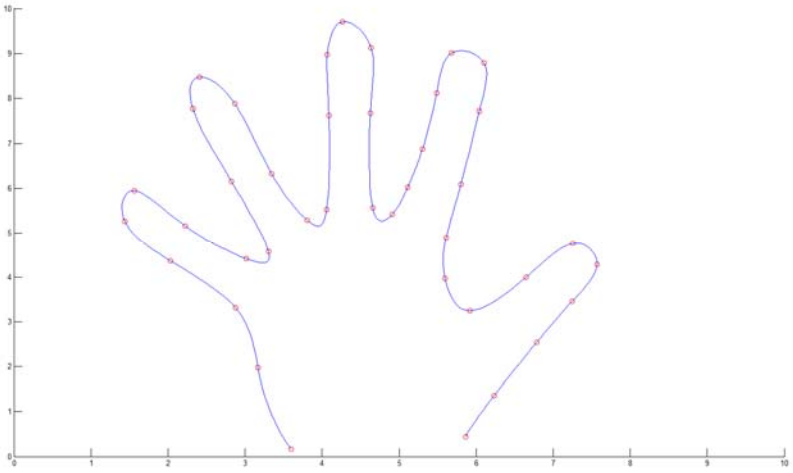
Diseño de curvas con splines cúbicos

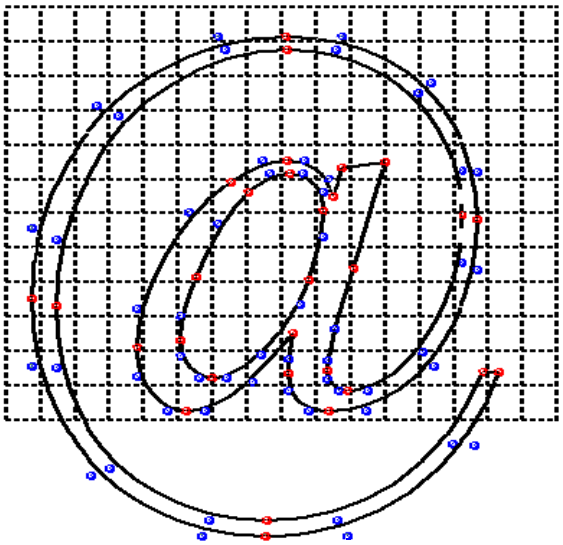
El objetivo de la práctica es dibujar curvas paramétricas utilizando splines cúbicos.
Dibujaremos curvas 1D (gráfica de funciones), 2D (caracteres, nombres) y 3D (diseño de una carretera de montaña).

Con una interface gráfica seleccionamos en pantalla una colección de puntos de control (nodos/valores). Interpolamos estos puntos de control con splines cúbicos y mostramos el resultado en pantalla.

Dibujar en pantalla vuestro **'perfil digital',** vuestro nombre y el perfil de la mano, y la letra @.







Indice:

Documentación y código:

En el curso Moodle tenéis las transparencias con el cálculo eficiente de splines cúbicos.

En el anexo de la hoja de entrega se incluye el código.

Pasos:

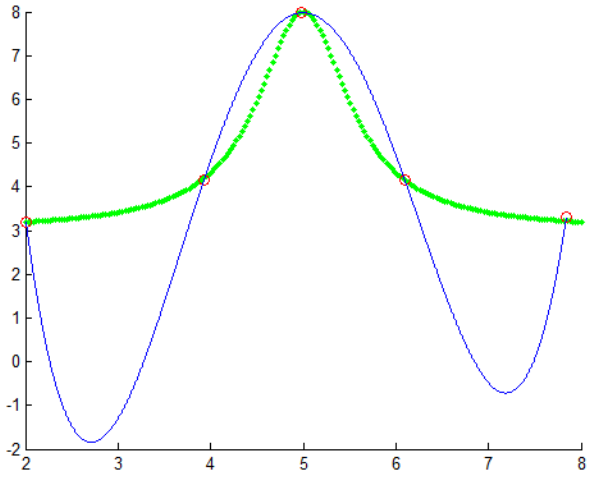
1. Dibujar una curva 1D. Interpolación de Newton.
Fenómeno de Runge: si aumentamos el número de nodos, el polinomio presenta oscilaciones.

```
>> DibujaCurva1D

% Arranca una interface gráfica.
% Con el botón de la izquierda del ratón seleccionamos los puntos/nodos (rojos) y con el botón de la derecha
% finaliza la selección de puntos, calcula el polinomio de Newton que los interpola (1D) y muestra el resultado (azul)
```

Seleccionar puntos de control sobre la curva verde que corresponde a la función $f(x) = \frac{1}{1+25x^2}$.

Al finalizar dibuja el polinomio de Newton que interpola esos puntos.



Añadir puntos de control hasta que el polinomio presente grandes oscilaciones. Adjuntar la gráfica.

Sustituir la función anterior por $g(x) = x(x-2\pi)e^{-x}$ en [0, 7]. Interpolar con 5 nodos con el mínimo error de interpolación posible. Adjuntar la gráfica.

2. Dibujar curvas 2D. Interpolación de Newton en cada una de las coordenadas de los puntos de control.
Dibujar caracteres: si el número de puntos de control es elevado el polinomio presenta oscilaciones.

```
>> DibujaCurva2D(nletras)        % nletras es el número de caracteres que quereis dibujar

% Arranca una interface gráfica.
% Con el botón de la izquierda del ratón seleccionamos los puntos/nodos (rojos) y con el botón de la derecha
% finaliza la selección de puntos, calcula el polinomio que los interpola (1D) y muestra el resultado (azul)
```

Seleccionar puntos de control para dibujar los caracteres $L \quad e \quad t \quad r \quad a$. Intentar que las curvas no tengan muchas oscilaciones.

3. Dibujar curvas 2D. Interpolación con un spline cúbico.
Dibujar curvas: caracteres, perfiles y firmas.

1. **Codificar la rutina auxiliar z= tri(r).**
2. **Codificar la rutina** yy=spline3(xi,yi,xx) **para calcular el spline cúbico**.
3. **En la rutina** DibujaCurva2D(nletras) **sustituir la función** polNewton **por la función** spline3.
4. **Dibujar curvas 2D con splines cúbicos: caracteres y perfiles.**

3.1. Codificar la rutina auxiliar z=tri(r).

La rutina tiene como variable de entrada el **vector columna r**, y como variable de salida el **vector z** (del mismo tamaño). El vector z es la solución del siguiente sistema lineal $Mz = r$ (se muestra un sistema de tamaña 6x6):

$$\begin{pmatrix} 1 & -2 & 1 & 0 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 & 0 \\ 0 & 1 & 4 & 1 & 0 & 0 \\ 0 & 0 & 1 & 4 & 1 & 0 \\ 0 & 0 & 0 & 1 & 4 & 1 \\ 0 & 0 & 0 & 1 & -2 & 1 \end{pmatrix} \begin{pmatrix} z_0 \\ z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{pmatrix} = \begin{pmatrix} r_0 \\ r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \end{pmatrix}$$

Como se puede observar, la matriz del sistema tiene el siguiente contenido:

- Las 3 primeras posiciones de la primera fila contiene los valores (1 -2 1). Las 3 últimas posiciones de la última fila contiene los valores (1 -2 1).
- Todas las demás filas k, en las posiciones k-1,k,k+1, contiene los valores (1 4 1).

Comprobar la rutina con:

```
>> r=[1:10]';z=tri(r)
z =
   0.7892
   0.1667
   0.5441
   0.6570
   0.8280
   1.0312
   1.0472
   1.7799
  -0.1667
   7.8868
```

3.2. Codificar la rutina yy=spline3(xi,yi,xx) **para calcular el spline cúbico**.

La rutina **yy=spline3(xi,yi,xx)** tiene como variables de entrada los vectores de nodos/valores xi/yi y el vector de valores xx donde se va a evaluar el spline cúbico, yy son las evaluaciones del spline en xx. Los nodos xi son equiespaciados.

Tabla de n+1 **nodos/valores xi/yi**:

$$xi : x_0, ..., x_n$$
$$yi : y_0, ..., y_n$$

Tabla de valores

$$xx = xi(1) : 0.01 : xi(end)$$

El **salto entre dos nodos** (equiespaciados) h: $h = xi(2) - xi(1)$

a) La rutina calcula los siguientes vectores:

El vector (de tamaño n+1) $F_0, ..., F_n$ donde:

$$F_0 = 0, F_n = 0$$

$$F_k = \frac{6}{h^2}\left(y_{k+1} - 2y_k + y_{k-1}\right) \quad \text{para } k = 1, ..., n-1$$

El vector (de tamaño n+1) $z = tri(F)$.

El vector (de tamaño n+1) $C_0, ..., C_n$ donde $C_k = y_k - \frac{h^2}{6} z_k$

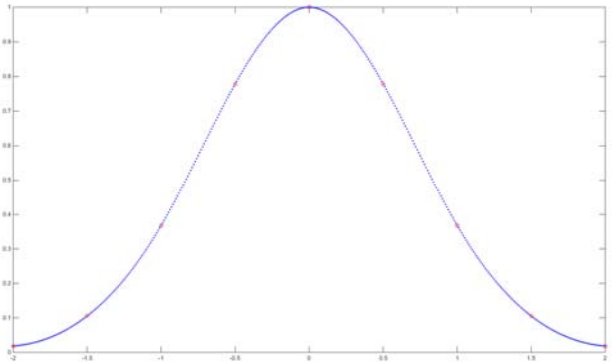
b) La rutina evalua el spline en un valor x, s(x). Codificar un bucle for para calcular s(x) para x=xx(1) hasta x=xx(end).

Calcular $w = \frac{x - x_0}{h}, \quad k = floor(w), \quad u = w - k$

El valor del spline en x, s(x), viene dado por

$$s(x) = \frac{h^2}{6} z_{k+1} u^3 + \frac{h^2}{6} z_k (1-u)^3 + C_{k+1} u + C_k (1-u)$$

Comprobar la rutina con:
>> xi=[-2:0.5:2]';yi=exp(-xi.^2);xx=-2:0.01:2;yy=spline3(xi,yi,xx);plot(xx,yy,'b.',xi,yi,'ro')



Calcular y dibujar el spline que interpola la función $f(x) = \frac{1}{1+25x^2}$ en el intervalo [-10,10] con 25 nodos equidistantes.

Nota 1: El enunciado se utiliza la notación usual de matemáticas, mientras que el código debe contener comandos Matlab.

$$\begin{cases} \text{Notación de matemáticas} \\ xi : x_0, ..., x_n \end{cases} \qquad \begin{cases} \text{Comandos Matlab} \\ N = length(xi);\ xi(1), ..., xi(N) \\ xi(1), ..., xi(end) \end{cases}$$

Nota 2: Para poder evaluar la rutina spline3 en el último nodo xi(end), necesitas extender las tablas C y z con un elemento nulo:
C(end+1)=0;z(end+1)=0;

3.3. En la rutina DibujaCurva2D(nletras) **sustituir la función** polNewton **por la función** spline3.

3.4. Dibujar curvas 2D con splines cúbicos: caracteres y perfiles.

Dibujar en pantalla el 'perfil digital' de los componentes del grupo: vuestro nombre y el perfil de vuestra mano.
Dibujar en pantalla la letra @ (similar a la de la gráfica).

Adjuntar las gráficas obtenidas.

4. Dibujar curvas 3D con splines cúbicos. Diseño de una carretera de montaña.

Ejecutar el código que se adjunta para obtener una serie de puntos de control (rojo) y la poligonal que une los puntos de control (azul).
Se desea construir una carretera de montaña que pase por los puntos de control. Para ello sustituir la poligonal por el spline que interpola los puntos de control.
Como se puede apreciar en la gráfica, el trazado de la carretera que se ha obtenido es suave excepto en un punto de control (P1). La curva en el punto P1 es de clase C^0 (continua) pero no es de clase C^1.
Calcular la poligonal de los puntos de control cambiando el orden: P10,P11,…P20,P1,P2,…P10.
Dibujar el spline que interpola esos puntos de control en el nuevo orden. El nuevo trazado de la carretera no es suave en otro punto de control.

Hacer las modificaciones necesarias para obtener una trazado de la carretera, que pase por todos los puntos de control y que sea suave en todos ellos.

```
clear all
P=zeros(21,3);
P(1,:)=[0 0 0.5];
P(2,:)=[0 1 -0.45];
P(3,:)=[0 2 0.4];
P(4,:)=[0 3 -0.35];
P(5,:)=[0 4 0.3];
P(6,:)=[0 5 -0.25];
P(7,:)=[0 6 0.2];
P(8,:)=[0 7 -0.15];
P(9,:)=[0 8 0.1];
P(10,:)=[0 9 -0.05];
P(11,:)=[1 9  0.05];
P(12,:)=[1 8 -0.1];
P(13,:)=[1 7  0.1];
P(14,:)=[1 6 -0.15];
P(15,:)=[1 5 0.15];
P(16,:)=[1 4 -0.2];
P(17,:)=[1 3 0.2];
P(18,:)=[1 2 -0.25];
P(19,:)=[1 1 0.25];
P(20,:)=[1 0 -0.3];
P(21,:)= P(1,:);
plot3(P(:,1),P(:,2),P(:,3),'ro', P(:,1),P(:,2),P(:,3),'b--');
```