

TAREA 1

Para el siguiente LAB necesitáis tener lista la función **interp_sat.m**, descrita en el ejercicio 3 de la 1ª hoja de LAB. Los ejercicios anteriores de esa hoja son los pasos necesarios para depurar y construir la función pedida. Para asegurarnos de que está lista la voy a pedir como tarea (por parejas).

En esta tarea **lo único que tenéis que subir a Moodle** es el correspondiente fichero m de MATLAB (interp_sat.m). Dicho fichero debe contener el código de vuestra función junto con el código de las funciones auxiliares pedidas (get_df, get_S). La organización del fichero será como sigue:

```
function [XYZ, cdT]=interp_sat(sp,tow,PRN,N)
% APELLIDOS1, NOMBRE1
% APELLIDOS2, NOMBRE2
... Código de la función interp_sat
return

function df=get_df(f)
% Función que recibe un vector f y devuelve vector df con
% las correspondientes diferencias finitas
... Código de vuestra función
return

function S = get_S(s,N)
% Recibe s y devuelve vector de coefs S para usar en interpolación
... Código de vuestra función
return
```

Si habéis usado alguna otra función auxiliar propia (para encontrar nodos, ...) añadidlas al final del fichero. Lo importante es que vuestra función pueda ejecutarse por su cuenta sin ningún fichero adicional. Para asegurarnos de ello, ponerla en un directorio vacío y comprobad que se ejecuta sin errores.

Para comprobar vuestra función os adjunto mi propia versión (mi_fun.p). Una función .p de MATLAB puede usarse como cualquier otra función, pero no permite que se vea su código. Su uso es idéntico al de interp_sat y si vuestra función es correcta os debe dar los mismos resultados.

ASPECTOS de la ENTREGA a TENER EN CUENTA (valorados en la nota)

- ENTREGAD solamente UN SOLO FICHERO interp_sat.m CONTENIENDO LAS TRES FUNCIONES (primero interp_sat y luego las auxiliares como se indica en el template anterior).
- Poner VUESTRO/S NOMBRE/S en las 2 primeras líneas de comentarios (usando %) y haced UNA SOLA ENTREGA por pareja.

- El resultado de la posición XYZ debe ser un vector columna (3x1), no fila (1x3)
- Antes de entregarla asegurarnos de que la función NO VUELQUE NADA POR PANTALLA: comentar los fprintfs, poner ; al final de las líneas para no volcar resultados, etc. Una llamada a la función:

```
[pos cdT]= interp_sat(sp,tow,prn,N);
```

no debería sacar nada por pantalla (el resultado se guarda en pos+cdT).

- Se valorará la velocidad del código (de ahí que no queramos hacerlo más lento volcando resultados intermedios por pantalla).
- Los datos GPS (estructura sp) pueden provenir de cualquier día de la semana, pero siempre estará garantizado que el tiempo tow donde se interpola no requerirá el uso de nodos de otros días.

Si completasteis el ejercicio 3 del LAB1, basta que adaptéis la función que ya tenéis a las especificaciones anteriores antes de entregarla.

La función anterior puntuará hasta un 75% de la nota de la tarea.

Para optar al 25% adicional tenéis que ampliar vuestra función para que también calcule las VELOCIDADES/DERIVA DEL RELOJ del satélite y devuelva esa información en un par de parámetros adicionales:

```
function [XYZ, dT, Vxyz, D] =interp_sat(sp,t,PRN,N)
```

Así, si llamamos a la función con: **[pos,cdT]=interp_sat(sp,t,PRN,n);**
devuelve la posición (pos) + error reloj (cdT) como antes (ambos en mt)

Si ahora la llamamos con: **[pos,cdT,vel,D] = interp_sat(sp,t,PRN,n);**
se obtienen adicionalmente la velocidad (vel) + deriva (D) (ambos en m/s).

Esta parte corresponde al ejercicio 5 de la hoja anterior. En dicho ejercicio se explican los detalles de la implementación y se dan algunos resultados para que podáis verificar vuestra función.

La implementación de esta parte no es necesaria para el algoritmo básico de posicionamiento (que no usa las velocidades de los satélites). Las velocidades se usarán si estamos interesados en mejorar los resultados implementando una pequeña corrección relativista que afecta al error de reloj del satélite en función de su velocidad.