

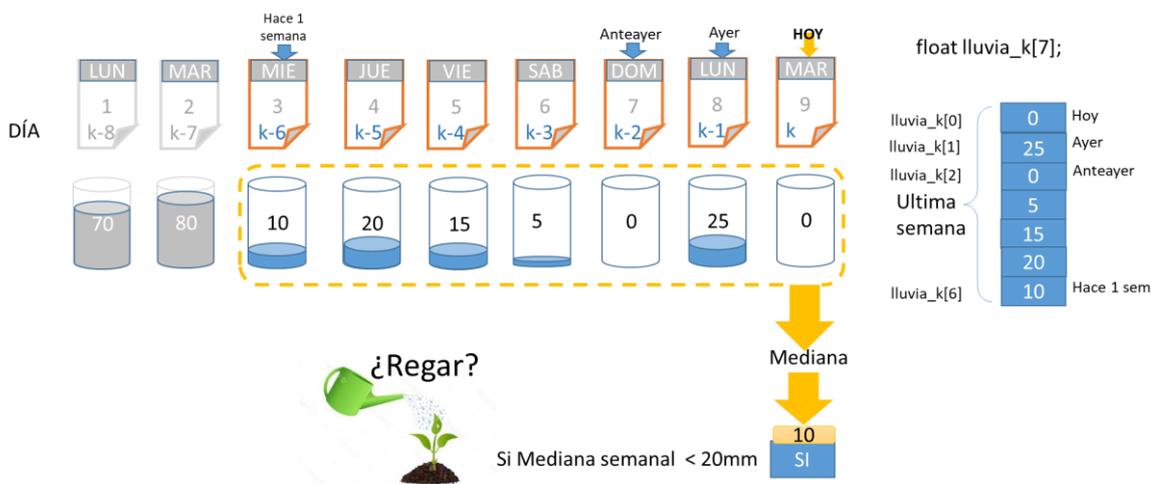


## Guía de Prácticas

ASIGNATURA: Informática Industrial y Comunicaciones  
CENTRO: Escuela Politécnica de Ingeniería de Gijón  
ESTUDIOS: Grado en Ingeniería Electrónica y Automática  
CURSO: 3º CUATRIMESTRE: 1  
CARÁCTER: Obligatoria CRÉDITOS ECTS: 6  
PROFESORADO: Ignacio Alvarez, José M<sup>a</sup> Enguita, Angel Navarro, Mariam Saeed

PRACTICA 05: Manejo de tablas (arrays) y series temporales

1. Ejercicio a realizar: programa que obtenga una medida cada día de un sensor de lluvia acumulada (se simulará mediante entrada de teclado), y decida cada día si hay que realizar el riego en función del valor de la mediana de las 7 medidas más recientes (última semana).



El programa tendrá el estilo siguiente

(ver [http://isa.uniovi.es/~ialvarez/Curso/descargas/series temporales regando.pdf?version=2024](http://isa.uniovi.es/~ialvarez/Curso/descargas/series_temporales_regando.pdf?version=2024)):

```
float lluvia_k[7];
float median_k;

InitTabla(lluvia_k, 7);
while (1)
{
    DesplazarTabla(lluvia_k, 7);
    lluvia_k[0]=LecturaSensor(); // realizar con scanf( )
    median_k=MedianaTabla(lluvia_k, 7);
    UtilizarResultado(median_k); // if (...) -> printf( )
    // Faltaría Sleep(Tm), pero no es necesario (se para en scanf)
}
```

- Realizar las funciones `InitTabla()`, `DesplazaTabla()` y `PrintTabla()`, y comprobar que en cada pasada del bucle (nuevo día) se obtiene la tabla con los datos correctos. Resultado esperado:

```

Introd valor lluvia: 70.↵
La tabla contiene: 70 0 0 0 0 0
Introd valor lluvia: 80.↵
La tabla contiene: 80 70 0 0 0 0
Introd valor lluvia: 10.↵
La tabla contiene: 10 80 70 0 0 0
...
Introd valor lluvia: 5.↵
La tabla contiene: 5 15 20 10 80 70
Introd valor lluvia: 0.↵
La tabla contiene: 0 5 15 20 10 80 70
Introd valor lluvia: 25.↵
La tabla contiene: 25 0 5 15 20 10 80
Introd valor lluvia: 0.↵
La tabla contiene: 0 25 0 5 15 20 10
...

```

- Obtención de mediana: ordenar una tabla de menor a mayor, tomar el valor central. Para ordenar la tabla: buscar el índice donde se produce el valor mínimo, intercambiar el valor de ese índice con el 1º elemento de la tabla; repetir lo mismo para la tabla que empieza en el 2º elemento y tiene n-1 datos, etc.

Ejemplo:  $t \equiv \{1.0, 3.5, 2.4, 1.7, -3.8\} \rightarrow ordenada \equiv \{-3.8, 1.0, 1.7, 2.4, 3.5\} \rightarrow mediana = 1.7$

Ordenación de tabla (necesita tabla auxiliar para no modificar la original).

ind	tam		i_min	tabla_ordenada				
				t+0	t+1	t+2	t+3	t+4
0	5	Tabla desde t+i ind y con tam elems →	4	1.0	3.5	2.4	1.7	-3.8
		Intercambia i_min y 1º para dicha tabla	→	-3.8	3.5	2.4	1.7	1.0
1	4	Tabla desde t+i ind y con tam elems →	3	-3.8	3.5	2.4	1.7	1.0
		Intercambia i_min y 1º para dicha tabla	→	-3.8	1.0	2.4	1.7	3.5
2	3	Tabla desde t+i ind y con tam elems →	1	-3.8	1.0	2.4	1.7	3.5
		Intercambia i_min y 1º para dicha tabla	→	-3.8	1.0	1.7	2.4	3.5
3	2	Tabla desde t+i ind y con tam elems →	0	-3.8	1.0	1.7	2.4	3.5
		Intercambia i_min y 1º para dicha tabla	→	-3.8	1.0	1.7	2.4	3.5
4	1	Tabla desde t+i ind y con tam elems →	0	-3.8	1.0	1.7	2.4	3.5
		Intercambia i_min y 1º para dicha tabla	→	-3.8	1.0	1.7	2.4	3.5

Tras finalizar la ordenación, la mediana es el valor que se queda en  $t[2] = 1.7$

- Añadir el uso de un contador para evitar cálculos incorrectos antes de disponer de suficientes datos:

```

float lluvia_k[7];
float median_k;
int counter=0;

InitTabla(lluvia_k,7);
while (1)
{
    DesplazarTabla(lluvia_k,7);
    lluvia_k[0]=LecturaSensor(); // realizar con scanf()
    counter++;
    if (counter>=7) {
        counter=7;
        median_k=MedianaTabla(lluvia_k,7);
        UtilizarResultado(median_k); // if (...) → printf()
    }
    // Faltaría Sleep(Tm), pero no es necesario (se para en scanf)
}

```

## RECORDATORIO:

Manejo de datos de series temporales: (diapositivas 49 y 56 de

[http://isa.uniovi.es/~ialvarez/Curso/descargas/Programacion%20de%20Control%20\(Bases\).pdf](http://isa.uniovi.es/~ialvarez/Curso/descargas/Programacion%20de%20Control%20(Bases).pdf)

Ejemplo de riego:

[http://isa.uniovi.es/~ialvarez/Curso/descargas/series\\_temporales\\_regando.pdf](http://isa.uniovi.es/~ialvarez/Curso/descargas/series_temporales_regando.pdf)

## AMPLIACIONES PROPUESTAS:

1. Solicitar por teclado al principio del programa el número de datos “a recordar”, para que no sean 7 de forma estática, y utilizar asignación dinámica de memoria para la tabla (malloc).
2. Añadir asignación dinámica de memoria para la tabla copia que se necesita crear en la función Mediana(): malloc() y free()
3. Añadir el algoritmo de decisión siguiente:

Si la mediana de lluvia de la última semana es menor que 10 y no se ha regado ninguno los dos días anteriores → regar

En caso contrario (mediana mayor o igual que 10, o bien se ha regado alguno de los dos días anteriores) → no regar.

Se requiere una segunda tabla para los resultados de riego, que se trata de forma similar a la de lluvia:

```
float lluvia_k[7]; // Se necesita 1 semana según el algoritmo
float riego_k[3]; // Se necesitan 3 días (actual y 2 anteriores)
...

InitTabla(lluvia_k,7);
InitTabla(riego_k,3);
while (1)
{
    DesplazarTabla(lluvia_k,7);
    DesplazarTabla(riego_k,3);
    lluvia_k[0]=LecturaSensor(); // realizar con scanf( )
    counter++;
    if (counter>=7)
    {
        counter=7;
        median_k=MedianaTabla(lluvia_k,7);
        riego_k[0]=CalcularRiego(median_k,riego_k[1],riego_k[2]);
        printf estado de riego_k[0]
    }
    // Faltaría Sleep(Tm), pero no es necesario (se para en scanf)
}
```

4. Solicitar por teclado al principio del programa el tipo de cálculo a realizar, que puede ser uno de los caracteres: ‘m’ (mediana), ‘>’ (máximo), ‘<’ (mínimo). Aplicar el cálculo indicado en lugar de la mediana.
5. Mejorar el algoritmo de cálculo de la mediana mediante la estrategia de división descrita en [http://sarielhp.org/research/CG/applets/linear\\_prog/median.html](http://sarielhp.org/research/CG/applets/linear_prog/median.html). Se requiere usar la función de C rand( ), que devuelve un entero aleatorio entre 0 y RAND\_MAX (para obtener un entero aleatorio entre 0 y n, usar rand( ) % (n+1)).