

Ejemplo de uso completo: crear un objeto `MySignal` a partir del número de datos indicado en el campo `<NumDatos>` de un texto xml:

Consola (pantalla + teclado)

```
Introduzca texto xml: esto es una prueba <NumDatos> 4 </NumDatos> ↵
Resultado: Creada una tabla MySignal con tamaño max = 4
—
```

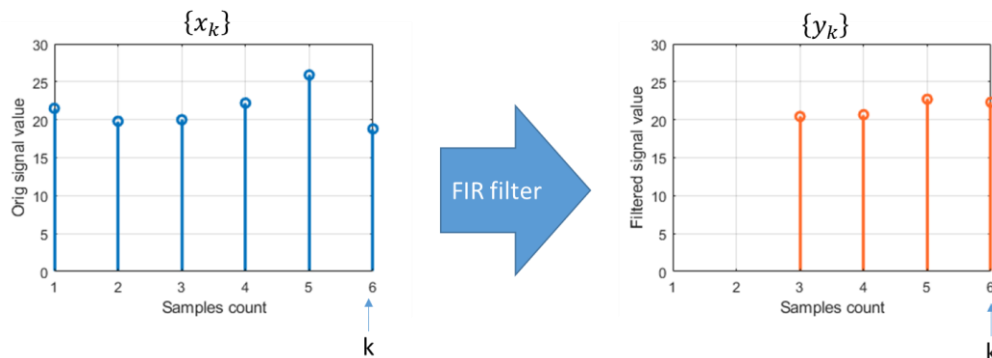
- Añadir una función para buscar y extraer de una cadena de caracteres el texto correspondiente a un tag xml y convertirlo a un valor real.

```
float XmlGetFloat(const QString& xml_str, const QString& xml_tag, bool& found);
```

- Añadir una función para buscar y extraer de una cadena de caracteres el texto correspondiente a un tag xml y convertirlo a una tabla de valores reales.

```
QList<float> XmlGetList(const QString& xml_str, const QString& xml_tag, bool& found);
```

- Realizar un programa que obtenga un vector de coeficientes de un texto XML y lo utilice para realizar un filtro FIR sobre los datos más recientes de temperatura adquiridos de un sensor:



FIR filter

$$y_k = b_0 * x_k + b_1 * x_{k-1} + b_2 * x_{k-2} = \sum_{i=0}^{m} (b_i * x_{k-i})$$

Un ejemplo: filtro FIR de media

Consola (pantalla + teclado)

```
Introduzca texto xml: <Coefs> [0.3333,0.3333,0.3333] </Coefs> ↵
Resultado:
Creada una tabla QVector<float> con datos [0.3333,0.3333,0.3333]
Creada una tabla MySignal con tamaño max = 3

Introduzca valor Temperatura: 21.5 ↵
La tabla contiene: 21.5 (incompleta, no se calcula filtro)
Introduzca valor Temperatura: 19.8 ↵
La tabla contiene: 19.8 21.5 (incompleta, no se calcula filtro)
Introduzca valor Temperatura: 20 ↵
La tabla contiene: 20.0 19.8 21.5 (avg = 20.433 )
Introduzca valor Temperatura: 22.2 ↵
La tabla contiene: 22.2 20.0 19.8 (avg = 20.667)
—
```

5) Tarea adicional:

Hacer lo mismo, usando `QFile`, para obtener los datos de un archivo de texto con formato XML y generar la salida de resultados hacia un archivo de texto cargable desde Matlab.

FileIn.xml

```
<Coefs>[0.3333,0.3333,0.3333]</Coefs>
<DataIn>[21.5,19.8,20,22.2,25.9,18.8]</DataIn>
```

FileOut.txt

```
21.5 0.0
19.8 0.0
20.0 20.433
22.2 20.667
25.9 22.700
18.8 22.300
```

Usar los argumentos de main (`argc`, `argv`) para indicar los nombres de los archivos de entrada y salida:

main.cpp

```
...
int main(int argc, char* argv[])
{
    // argc = Núm de argumentos
    //         (incluido nombre de programa)
    // argv[i]=Strings con los argumentos
    // argv[0]=Nombre del ejecutable
    // argv[1]=Primer argumento
    // ...
    // argv[argc-1]=Ultimo argumento
}
```

