

Contenidos Informática II

Curso R2004
UTN - FRBA

Introducción

- Empezaremos el año presentando un nuevo lenguaje de programación, que es una evolución del C (o sea, que permite programar en C, pero agrega nuevas características). Este lenguaje es C++, y aprovecharemos sus características para hablar de la Programación Orientada a Objetos (POO).
- Introduciremos posteriormente herramientas para trabajar con varios procesos en paralelo y comunicarlos.
- Finalmente, aprovechando estas herramientas, desarrollaremos interfaces gráficas, que son programas en C++ que permiten al usuario comunicarse con la aplicación mediante gráficos, botones, y otros objetos que denominaremos WIDGETS.

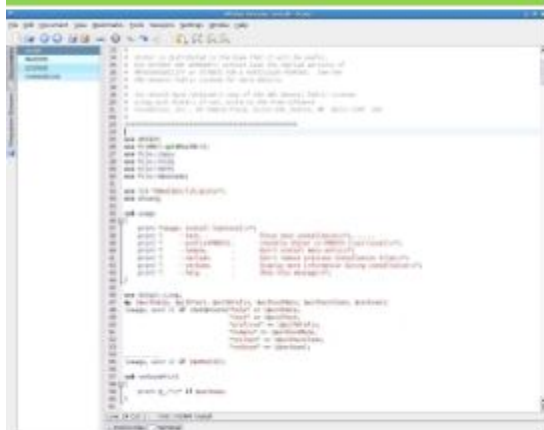
Introducción (II)

- Luego investigaremos cómo diseñar un programa, independientemente de su lenguaje, utilizando una herramienta que denominamos máquinas de estado. Esta herramienta es especialmente útil cuando desarrollamos programas secuenciales, típicos de los sistemas de automatización y control en la industria, aunque sirven para varios usos. Luego probaremos distintas formas de codificar estos diseños, siempre en lenguaje C.
- Finalmente, la segunda mitad del año incursionaremos en el desarrollo de programas para sistemas embebidos. Esto implica que nuestras aplicaciones ya no serán ejecutadas por la PC, sino que las bajaremos a un microcontrolador (LPC845), e intentaremos controlar los periféricos del mismo, utilizando las estrategias de diseño del tema anterior (máquinas de estado) y configurando cada periférico de acuerdo a lo que nos indica el manual del microcontrolador

En forma gráfica...

En Informática I

Editor de texto



Compilador

GCC

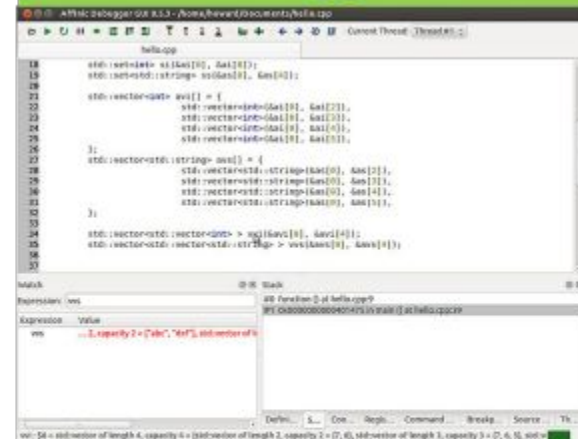
bibliotecas

Estandar
OpenCV
OpenGL
etc.

Linker

GCC

Debugger



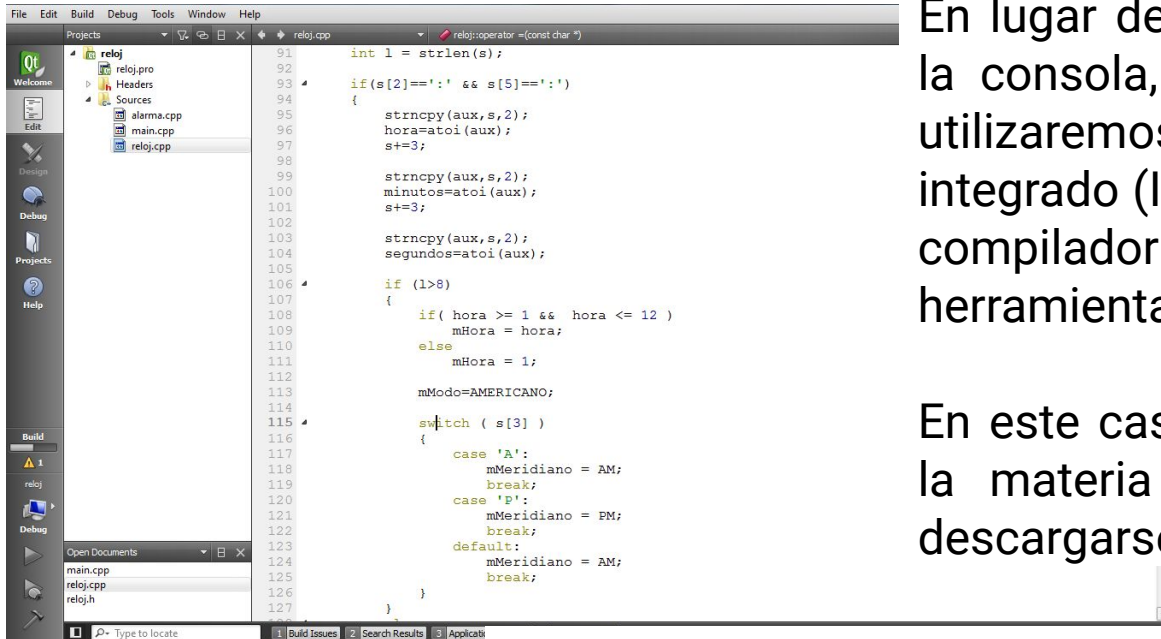
¡¡Todo suelto!!!

En forma gráfica...

En Informática II

En lugar de escribir nuestros programas en la consola, linkearlos, compilarlos a mano, utilizaremos un entorno de desarrollo integrado (IDE), en donde tenemos un editor, compilador y linker, más algunas herramientas adicionales.

En este caso, el entorno que elegimos para la materia es el Qt Creator, que puede descargarse de <http://www.qt.io/download>



Para la próxima clase, les pedimos que bajen e instalen el IDE (la descarga es gratuita), ya que lo utilizaremos para desarrollar los primeros ejemplos

C++

Comenzaremos el año cambiando el lenguaje de programación y aprendemos lo que son los OBJETOS, y que estrategias de programación pueden aprovecharse con este nuevo lenguaje:

```
#ifndef CLASS_STRING_H
#define CLASS_STRING_H

#include <iostream>
#include <cstring>

using namespace std;

class String
{
    char *p;
    int len;
public:
    String();
    String(const char *);
    String(const String&);
    ~String();

    String operator+(const String &s);
    String operator+(const char *s);
    String& operator=(const String &s);
    String& operator=(const char *s);

    friend String operator+(const char *s1, const String &s2);
    friend ostream& operator<< (ostream& , const String& );
};

String operator+(const char *s1, const String &s2);
ostream& operator<< (ostream &o, const String &s);

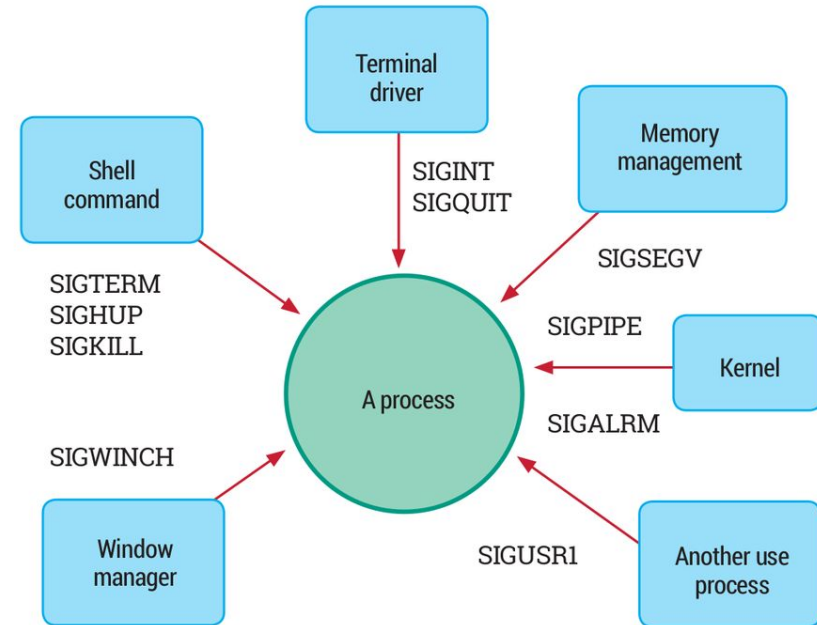
#endif // CLASS_STRING_H
```

Threads, IPCs

Utilizando el lenguaje C++ sobre el IDE QtCreator, destinamos las siguientes clases a generar PROCESOS, THREADS (hilos de ejecución), y analizar como los comunicamos entre si:

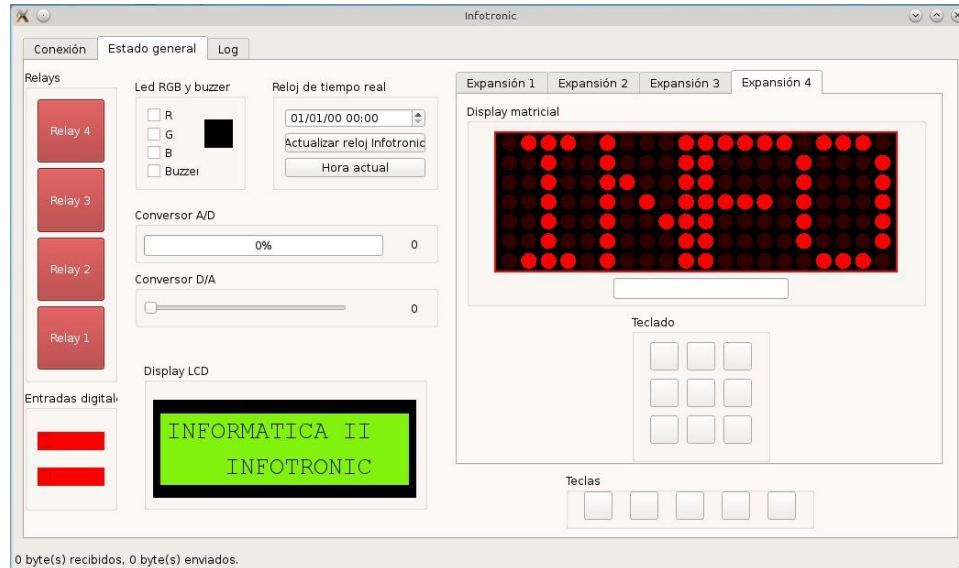


Comunicación entre procesos



Interfaces gráficas

El Qt Creator, además de soportar los lenguajes C y C++ (no deberemos de cambiar la metodología de trabajo al cambiar el lenguaje), nos permitirá además programar interfaces gráficas, pudiendo entonces diseñar aplicaciones como la que vemos abajo:



Programación de sistemas embebidos (SE)

En la segunda mitad del año, introduciremos el concepto de microcontrolador, periféricos, y sistemas embebidos, y aprenderemos a generar aplicaciones que se ejecuten en una placa de desarrollo que tendremos para tal fin:



Descarga del programa

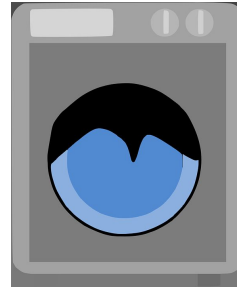


Durante la segunda mitad del año, utilizaremos un nuevo IDE: El MCUXpresso, que nos permitirá, además de editar, linkear y compilar el programa, bajarlo al SE y DEBUGGEARLO en tiempo real



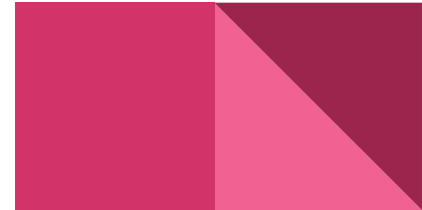
Aplicaciones para Sistemas embebidos

Ejemplos de sistemas embebidos:



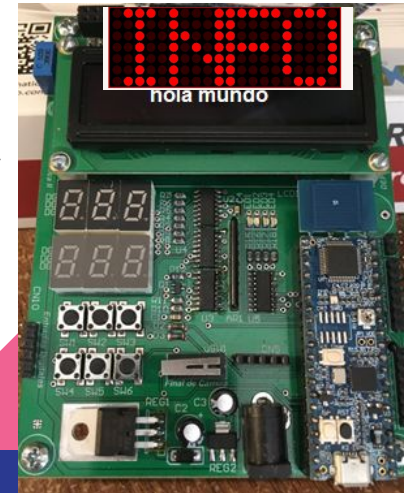
En general, podemos pensar un sistema embebido como un circuito electrónico inteligente que se encuentra diseñado y programado para cumplir una tarea específica

Como trabajo práctico obligatorio de la materia, deberemos desarrollar a lo largo del año (en grupos de 4 personas) una aplicación que se ejecute en un sistema embebido.



Trabajo práctico obligatorio de la materia (TP0)

El trabajo práctico obligatorio consistirá en desarrollar un programa que se ejecute en el SE, y en paralelo una interfaz gráfica que se ejecute en la PC, y lograr que se comuniquen efectivamente, de manera de poder controlar el sistema embebido desde una PC via WiFi.



Bibliografía

- LPC84x User manual, File name: UM11029 (bibliografía gratuita descargable del sitio
<https://studio.segger.com/packages/LPC800/CMSIS/Documents/UM11029.pdf>)
- Deitel/Deitel(2004)–Cómo programar en C/C++ -Prentice Hall–4ta.edición–Español
- Fatos Xhafa/Pere Pau Vazquez Alcocer/otros (2006) – Programación en C++ para Ingenieros – Thompson – 1ra. Edición
- Kernighan, B./Ritchie, D. (1991) – El Lenguaje de Programación C –Prentice Hall.