



VNiVERSiDAD
D SALAMANCA

Conecta 4

Computadores II

Samuel Gómez Sánchez, 45136357-F

Miguel Díaz Galán, 72898082-L

1º Grado en Ingeniería Informática

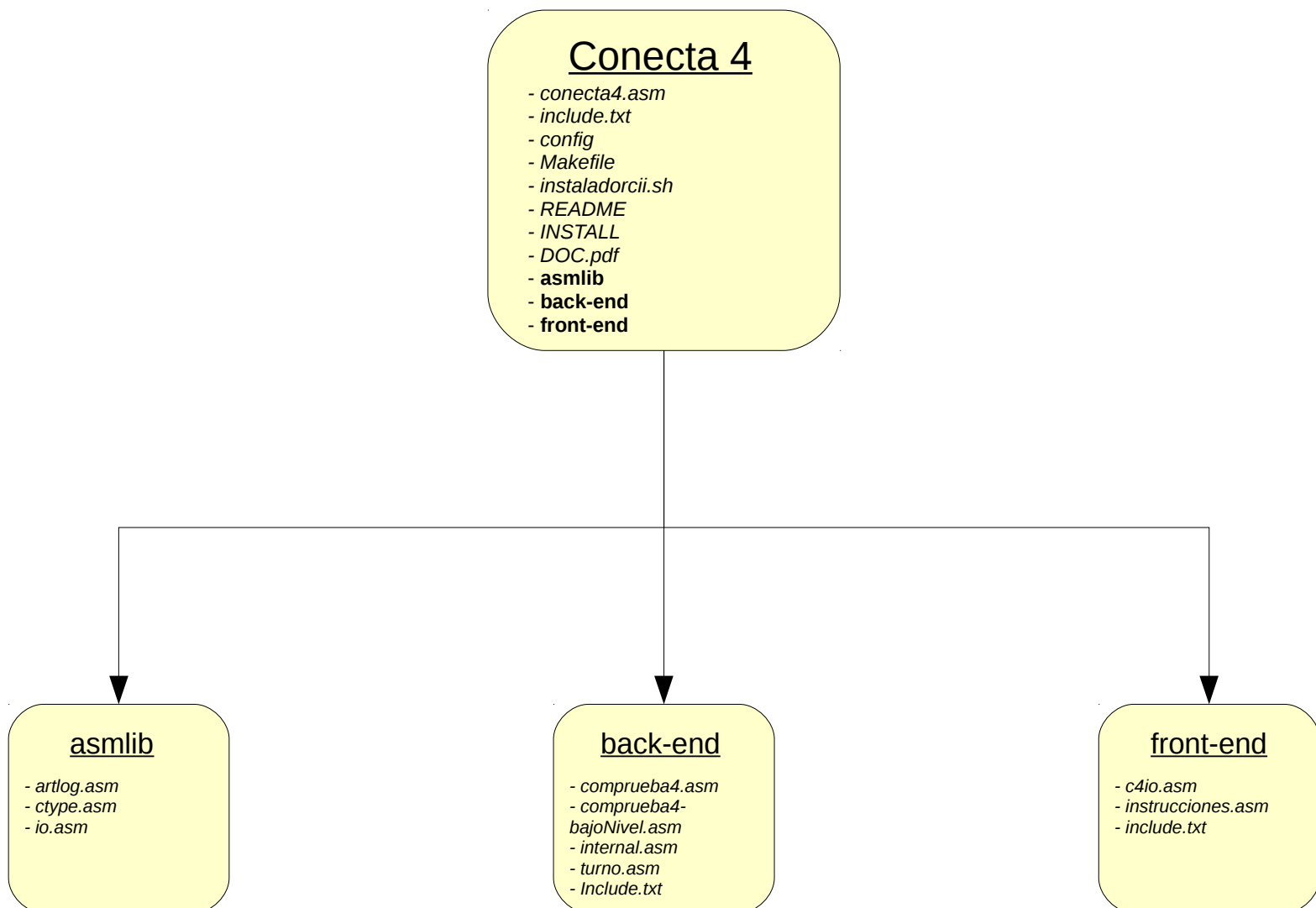
Conecta 4

Introducción

Conecta 4 es un juego desarrollado en lenguaje ensamblador del procesador Motorola 6809, versión digital del juego original de Hasbro (https://es.wikipedia.org/wiki/Conecta_4).

Árbol de directorios

El proyecto del juego Conecta 4 se estructura del modo siguiente:



Directorio principal

Contiene el módulo principal y los ficheros de configuración del entorno e instalación.

- *conecta4.asm*: módulo principal que ejecuta el programa núcleo.
- *include.txt*: contiene directivas para objetos compartidos entre módulos. Está incluido en el resto de módulos como una réplica exacta para poder trabajar con él.
- *config*: script con prompt de configuración del entorno.
- *instaladorcii.sh*: script obtenido de la página de la asignatura, avellano.usal.es/~compil, que instala los programas *as6809*, *aslink* y *m6809-run*, que permiten la emulación del procesador Motorola 6809 para el que se ha desarrollado el programa.
- *INSTALL*: instrucciones de instalación rápida gracias a la herramienta *make*.
- *README*: información breve para el usuario. **Empezar leyendo este fichero** para saber cómo proceder.
- *DOC.pdf*: este fichero.

asmlib

Directorio con subrutinas genéricas para el ensamblador del 6809. Contiene los módulos:

- *artlog.asm*: subrutinas de operaciones aritmético-lógicas.
- *io.asm*: subrutinas de entrada/salida.
- *ctype.asm*: subrutinas para el tratamiento de caracteres.

back-end

Contiene módulos con subrutinas que implementan la lógica interna del programa.

- *internal.asm*: subrutinas de operaciones genéricas en el conjunto del juego.
- *turno.asm*: subrutinas para la ejecución de las operaciones necesarias durante el turno en una partida, como determinar qué jugador debe jugar a continuación.
- *comprueba4-bajoNivel.asm*: subrutinas de bajo nivel para el análisis del tablero en busca de posición de 4 en raya.
- *comprueba4.asm*: subrutinas que utilizan las contenidas en *comprueba4-bajoNivel.asm* para llevar a cabo todos los test necesarios para ver si existe posición de 4 en raya.
- *include.txt*: cf. "Directorio principal"

front-end

Contiene módulos con subrutinas que implementan la interfaz de usuario.

- *c4io.asm*: contiene las operaciones de entrada/salida necesarias durante la ejecución del programa, a excepción de la presentación de las instrucciones.
- *instrucciones.asm*: contiene el código necesario para la presentación de las instrucciones al usuario.
- *include.txt*: cf. "Directorio principal"

Para más información, cada fichero contiene una cabecera explicativa, así como cada subrutina posee una presentación clara de las operaciones que realiza y sus parámetros de entrada y salida.

Notas adicionales

Dado que el desarrollo del proyecto se ha realizado en paralelo al aprendizaje del lenguaje, existen mejoras evidentes a gran parte del código, que con tiempo para llevarlas a cabo supondrían una mejora en seguridad, eficiencia, coherencia y consistencia del mismo. Entre otras, cabe señalar:

- Eliminación de etiquetas de variables locales a subrutinas y sustitución por uso de pila; sustitución de saltos de tipo absoluto por relativos; intercambio de direccionamientos inmediatos por relativos al PC; todo ello para permitir reubicabilidad en el código.
- Mejora en el restablecimiento del estado del microprocesador tras la llamada a subrutinas: algunas modifican el contenido de determinados registros (en especial, el de códigos de condición) de manera innecesaria.
- Reestructuración de determinadas capas de software para evitar redundancias y otras operaciones innecesarias.