

Arquitectura de Computadoras



Trabajo Práctico Especial: Manual de Usuario

Grupo: 03

Integrantes:

- Pablo Gorostiaga (65148)
- Carlos Amador Vallejo Tapia (65454)
- Tiago Heras (65627)

1. Requisitos del Sistema:

Para utilizar la shell de este kernel, es necesario contar con:

- QEMU: Entorno de emulación compatible con Linux, MacOS o Windows (WSL).
- Docker: Necesario para la compilación.

2. Instalación y Configuración:

Descarga Docker:

```
docker pull agodio/itba-so:2.0
```

Creación del Contenedor Docker:

```
docker run -d -v ${PWD}:/root --security-opt  
seccomp:unconfined -it --name TPE-Arqui  
agodio/itba-so:2.0
```

Compilación y Ejecución:

Para ejecutar:

[./run.sh](#)

Si se desea compilar y luego ejecutar:

[./compAndRun.sh](#)

3. Comandos en la Shell:

Nuestra shell cuenta con los siguientes comandos:

1. Help:

El comando 'help' muestra en pantalla todos los comandos implementados en la shell, así como una breve descripción de la funcionalidad de cada comando.

2. Time:

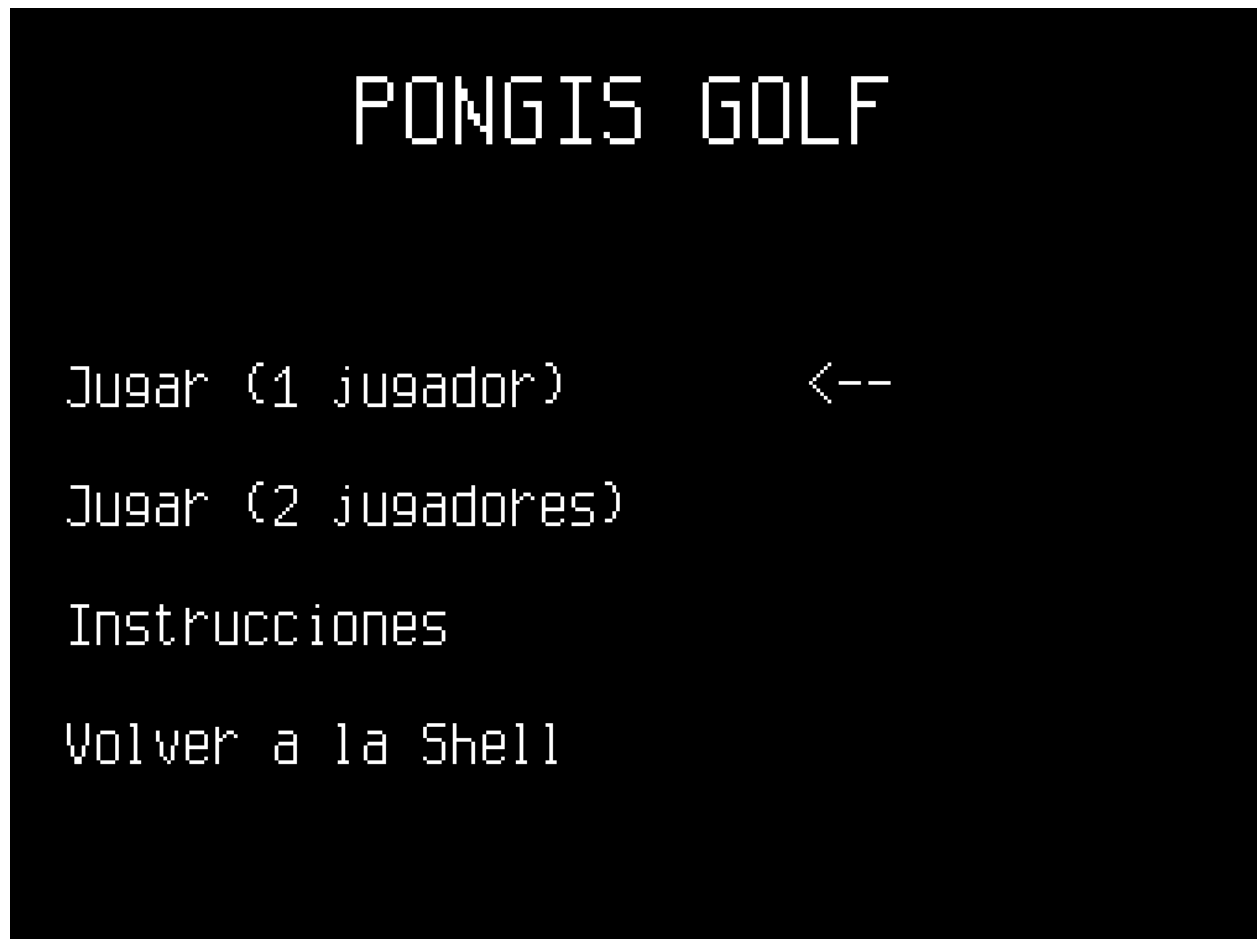
El comando 'time' exhibirá la hora en tiempo real para la zona horaria de Argentina (GMT-3).

3. Registers:

El comando 'regs' mostrará el contenido de todos los registros justo antes de que se ejecute la instrucción.

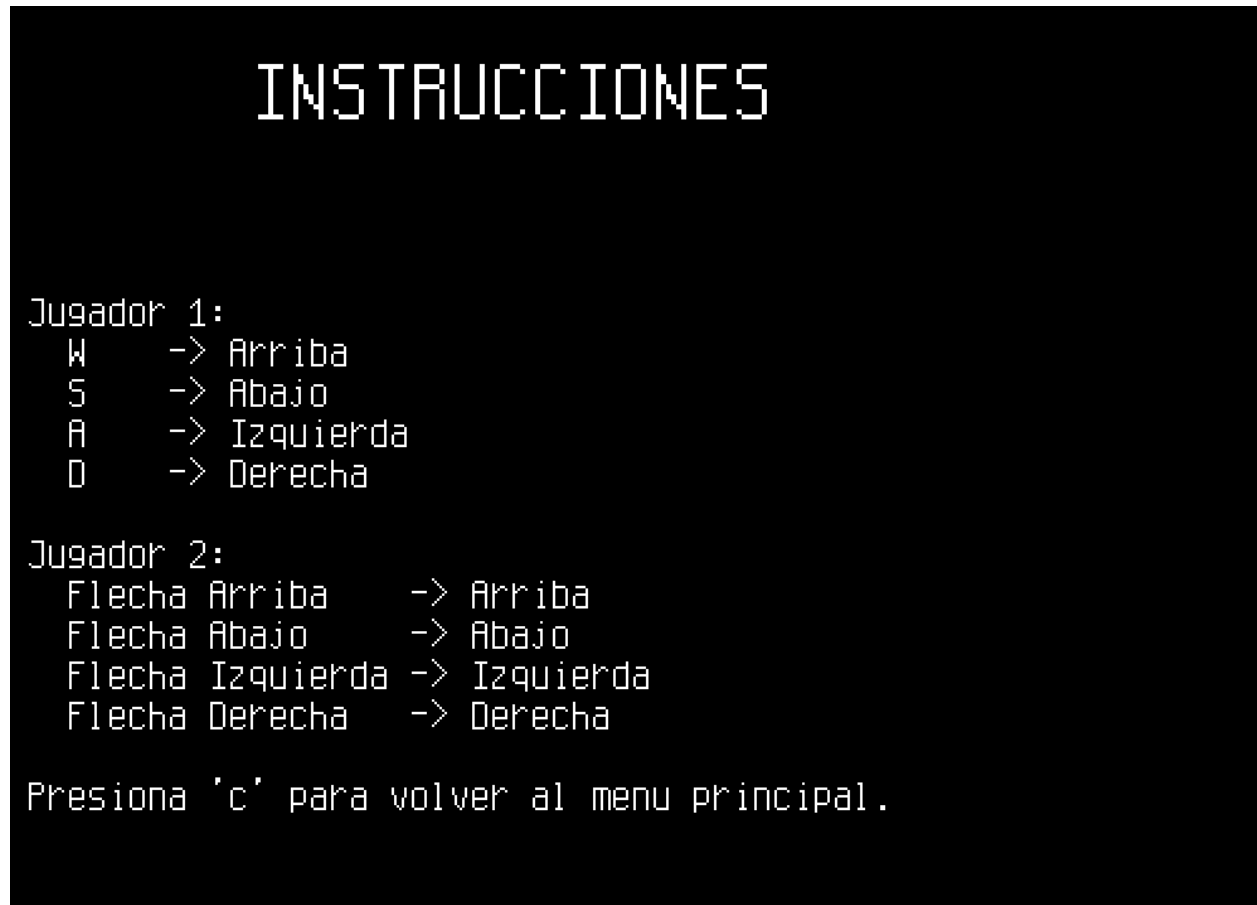
4. Pongis:

El comando 'pongis' abrirá el menú del juego Pongis-Golf. Aquí se mostrarán 4 opciones, por las cuales se puede desplazar utilizando las teclas 'w' (Para moverse hacia arriba) y 's' (Para moverse hacia abajo), y se puede seleccionar la opción deseada mediante la tecla ENTER. Se mostrarán las siguientes opciones:



1. Jugar (1 jugador): Empezará el juego en el primer nivel con 1 jugador.
2. Jugar (2 jugadores): Empezará el juego en el primer nivel con 2 jugadores.

3. Instrucciones: Mostrará los controles pertinentes al juego. Los cuales son los siguiente:



4. Volver a la Shell: Retornara a la Shell.

El juego consiste en meter la pelota (Azul) en el hoyo (Gris). Se puede mover la pelota mediante la colisión entre jugador-pelota.

5. Division by Zero:

El comando 'div0' ejecuta una excepción de tipo "cero división". Muestra el contenido de los registros cuando ocurre la excepción. Sirve para testear cómo se vería la excepción en caso de ocurrir en otro momento.

6. Invalid Operation:

El comando 'invalidOp' ejecuta una excepción de tipo "invalid opcode". Muestra el contenido de los registros cuando ocurre la excepción. Sirve para testear cómo se vería la excepción en caso de ocurrir en otro momento.

7. Zoom:

El comando 'zoom <NUMBER>', donde <NUMBER> es un argumento ingresado por el usuario, modifica el tamaño del texto de la terminal. El valor de <NUMBER> puede ser entre 1 y 10, cualquier otro valor será ignorado y mostrará un mensaje de error en pantalla.

8. Exit:

El comando 'exit' cierra el emulador.