



Instituto Tecnológico
de Buenos Aires

Manual de Usuario “Titanium OS”

Alumnos:

Dorado Tomás, 56594.

Dallas Tomás, 56436.

Fecha de entrega TPE: 5-6-2018.

Fecha de defensa TPE: 6-6-2018.

TPE - Arquitectura de Computadoras

Documentación sobre **TITANIUM OS**

Ésta es la documentación final sobre la versión 1.0 de TITANIUM OS. Una implementación basada en **PURE64** (material provisto por la cátedra) y modificado para los casos de uso solicitados.

1.Descarga e instalación

- **1.1** Descarga
- **1.2** Instalación

2.Primer ejecución y casos de uso básicos

- **2.1** Primera ejecución
- **2.2** Comandos

3. Manejo de excepciones

- **3.1** Forzado de excepciones
- **3.2** Manejo de excepciones
- **3.3** Vuelta a la terminal

4. BlobWars

- **4.1** Información sobre el juego
- **4.2** Implementación
- **4.3** Modos de juego

5.Reporte de bugs

1. Descarga e instalación

1.1. Descarga

1) Ejecutar el siguiente comando en una terminal de linux:

```
git clone https://github.com/tdallas/arqui-tpe
```

En el caso de poseer el código fuente en formato .zip, descomprimir el archivo en la carpeta dónde quiera contener el SO.

1.2. Instalación

2) Preparar el ambiente para poder ejecutar el SO. Para ello, ejecutar los siguientes comandos en una terminal de linux:

```
Instalar los paquetes necesario  
sudo apt-get install nasm qemu gcc make
```

Ir a la carpeta donde posee los códigos fuentes y luego buildear el la Toolchain

```
cd Toolchain  
make all
```

Buildear el kernel

```
make all
```

3) Ahora ya puede ejecutar nuestro SO, para ello, pararse en la carpeta donde se encuentra el código fuente y ejecutar en una terminal:

`./run.sh`

2. Primera ejecución y casos de uso básicos

2.1. Primera ejecución

Al ejecutarlo por primera vez, luego de ejecutar el último comando comentado en la sección 1.2. Se abrirá el emulador QEMU con el sistema funcionando.

Al bootear, se toma un delay de 2 segundos en el que se podrá escuchar dos pitidos con distintos tonos (no se asuste, es nuestro sistema diciendo sus primeras palabras!), se podrá ver la terminal esperando un comando para procesar.

2.2. Comandos

En nuestra terminal se podrán ejecutar varios comandos básicos, para ejecutar un comando, deberá escribirlo y presionar la tecla ENTER:

- Para visualizar información básica sobre el sistema, ejecute:

`info`

- Para visualizar ayuda sobre los comandos, ejecute:

`help`

- Para limpiar la pantalla (dejar solo el prompt), ejecute:

`clear`

- Para mostrar la hora en formato consola (texto plano), ejecute:

`displayTime -console`

- Para mostrar la hora en formato digital, ejecute:

displayTime -digital

Dentro de esta pantalla podrá modificar los colores del reloj mediante el hotkey c y volver a la terminal con el hotkey b o ESC. Siempre que interactúe mediante hotkeys, escuchará un pitido (es nuestro sistema hablándole a usted).

- Para modificar una zona horaria, ejecute el siguiente comando, donde NUMBER representa una zona horaria válida en formato de número:

setTimeZone NUMBER

- Para jugar al BlobWars, el cual se detallará en las próximas secciones, ejecute:

blobWars

- Para disparar una excepción de división por cero y ver cómo se maneja, ejecute:

exceptionZero

- Para disparar una excepción de operación inválida y ver cómo se maneja, ejecute:

exceptionOpCode

- Para salir del sistema, ejecute:

exit

- Para cambiar de color el fondo del sistema, donde COLOR puede ser default, red, orange, yellow, green o blue , ejecute:

changeTextColor COLOR

- Para cambiar el color del texto utilizado en la consola, donde COLOR puede ser default, red, orange, yellow, green o blue, ejecute:

changeBackgroundColor COLOR

- Para que la terminal imprima en pantalla lo que usted escriba, donde MENSAJE es cualquier mensaje que usted desee, ejecute:

echo MENSAJE

3. Manejo de excepciones

Se podrá probar el manejo de dos tipos de excepciones, la excepción que se produce cuando se divide por cero, y la división que se produce cuando se ejecuta una operación inválida.

3.1 Forzado de excepciones

Para probar nuestro manejo de excepciones, forzamos las mismas a través de código simple en assembler.

3.2 Manejo de excepciones

Para manejar el forzado de estas excepciones en el sistema, primero catcheamos la excepción, y mediante un printeo en pantalla detallamos qué tipo de excepción fue disparada, para luego volver a nuestro entorno de terminal y poder continuar con el flujo de nuestro sistema.

3.3 Vuelta a la terminal

En la vuelta a la terminal, se imprime nuevamente nuestro menú (para no perderlo de vista) y se podrá ver a nuestro prompt esperando instrucciones.

4. BlobWars

En el sistema podrán jugar este juego, el cual fue realizado por nosotros, para un TP de la materia Programación Imperativa.

Este juego fue realizado principalmente dándole un enfoque en cuanto a eficiencia, testado al máximo para prevenir memory leaks (pérdidas de memoria) y optimizado lo mejor posible.

4.1 Información sobre el juego

El juego consiste en un tablero cuadrado, donde se ubican fichas de dos jugadores distintos. Cada jugador, mediante movimientos válidos (1 posición a la redonda) debe intentar transformar las fichas del otro como suyas. Gana el jugador que más fichas obtiene.

4.2 Implementación

La interfaz realizada para este juego le pregunta en primera instancia al usuario la dimensión que quiere para su tablero (recordar que debe ser cuadrado), y luego se colocan fichas de manera aleatorias y se comienza el juego. El primer turno es asignado aleatoriamente, los comandos esperados para el movimiento de

las fichas es de forma [xx,xx][yy,yy], donde el primer término de cada corchete ([xx,xx][yy,xx]) hace referencia a la posición desde dónde se mueve, y el segundo término hasta dónde se realiza el movimiento deseado (recordar que el mismo debe ser válido).

4.3 Modos de juego

Se dispone de 3 modos de juego distintos:

1. 1 vs 1, donde juega un usuario1 contra otro usuario2.
2. 1 vs IA, donde juega un usuario contra el sistema. (Juega muy bien, cuidado)
3. IA vs IA, donde juegan el sistema contra el sistema. Divertida manera de ver cómo el sistema pierde y gana a la vez.

5. Reporte de bugs

En el caso de encontrar algún bug en el sistema nos gustaría tener conocimiento de ello para así poder solucionarlo. Para poder entender bien el contexto en el que el bug sucedió, nos gustaría que nos detalle cuales fueron los pasos que siguió anteriormente y que error generó en el sistema (se detuvo, se cerró, etc). Rogamos se comunique con nosotros vía mail a cualquiera de estas dos casillas con la información recolectada sobre el problema:

- tdallas@itba.edu.ar
- tdorado@itba.edu.ar