

Manual de Usuario

Trabajo Práctico Especial
Segundo Cuatrimestre 2017
Arquitecturas de Computadoras

- **Integrantes del grupo:** Lóránt Mikolás (57347), Johnathan Katan Piñeiro (56653) y Matías Heimann(57503).
- **Fecha de vencimiento de la entrega:** 14 de noviembre a las 23:59hs.

Instalación

En linux se debe contar con nasm, qemu, gcc y make.

1. Extraer en una carpeta el .zip recibido.
2. Abrir la carpeta en la terminal.
3. Moverse al directorio
TP-Arqui/RowDaBoat-x64barebones-d4e1c147f975/Toolchain/
4. Ejecutar comando **make all**
5. Luego en **TP-Arqui/RowDaBoat-x64barebones-d4e1c147f975/** ejecutar comando **make all**

En caso de que no funcione probar hacer **make clean** antes de **make all**.

Booteo en máquina real

Se debe seguir el instructivo proveído por la cátedra: “Para volcar la imagen generada a un pendrive es necesario correr la herramienta **dd**, que copia y transforma los datos, lo interesante es que puede leer y escribir de archivos que en realidad son dispositivos de almacenamiento.

Primero es necesario identificar la unidad de almacenamiento a utilizar. Para ello debe correr el comando **'fdisk -l'** que lista todos los dispositivos de almacenamiento y sus particiones. (Dependiendo del sistema, puede llegar a requerir de un super usuario) Las unidades terminadas en un número son en realidad particiones dentro del disco. Por ejemplo, /dev/sda1 es una partición del disco /dev/sda.

Se debe copiar en el disco, y no en la partición.

Una vez elegido el disco, correr el comando, donde **if** es input, y **of** es output “(…)”

```
root@debian:/home/anizzomc/x64barebones/Image# dd if=x64BareBonesImage.img  
of=/dev/sdb  
12288+0 records in  
12288+0 records out  
6291456 bytes (6.3 MB) copied, 26.1582 s, 241 kB/s
```

En sus máquinas dicho comando requiere de ser super-user. En el laboratorio, no hay peligro porque **dd** fue habilitada para modificar unidades de los usuarios.”

Por último se debe elegir como boot device previo al encendido el pendrive correspondiente.

Utilización sobre QEMU

Desde la terminal en la carpeta

TP-Arqui/RowDaBoat-x64barebones-d4e1c147f975/

ejecutar el comando:

make fromzero

Este comando, hace make clean, make all y ejecuta el tpe.

Comandos

Comandos de la shell

Los comandos disponibles son:

- **echo**: para imprimir por terminal un string.
- **clear**: limpia la pantalla y la pone en negro.
- **divide by cero**: genera la excepción de división por cero.
- **overflow**: genera la excepción de overflow.
- **invalid opcode**: genera la excepción de opcode inválido.
- **time**: muestra la información proveída por el real time clock. Presenta [la fecha actual y la hora actual](#).
- **help**: muestra los comandos disponibles.
- **graph**: ejecuta un programa para graficar funciones.

Tenga en cuenta que se debe presionar la tecla enter para ejecutarlos.

Comandos del programa para graficar funciones (PLOTTER PROGRAM)

Los comandos disponibles son:

- **addplot**: este comando sirve para graficar una función polinómica de grado menor o igual a dos. Luego de ejecutar el comando el usuario deberá ingresar por separado los coeficientes a, b y c. Seguidos por un enter.
- **exit**: cierra el programa y vuelve a la shell.

Syscalls

Las syscalls soportadas son:

- **write**: escribe en entrada estándar.
- **read**: lee de entrada estándar.
- **clearScreen**: limpia la pantalla.
- **paintPixel**: pinta un píxel en las coordenadas indicadas.
- **getResolutionY**: devuelve la resolución vertical.
- **getResolutionX**: devuelve la resolución horizontal.
- **printRTCInfo**: imprime la información del *real time clock*.

Syscall	RAX	RBX	RCX	RDX
write	3	uint64_t fd	uint64_t buffer	uint64_t count
read	4	uint64_t fd	uint64_t buffer	uint64_t count
clearScreen	5	-	-	-
paintPixel	6	uint64_t x	uint64_t y	-
getResolutionX	7	-	-	-
getResolutionY	8	-	-	-
printRTCInfo	9	-	-	-

Librería Estándar de User Space

Las funciones disponibles son:

- **printf** : se debe tener en cuenta que no puede recibir formatos para la impresión, como por ejemplo %d.
- **strlen**: simplemente cuenta la cantidad de caracteres de un string.
- **scanf**: al igual que printf, nuestra implementación de scanf no puede recibir formatos para el guardado de los datos leídos.
- **getChar**: es equivalente a la función getchar() de c.
- **putchar**: es equivalente a la función putchar() de c.
- **println**: sirve para imprimir enteros a la salida estándar.
- **countDigits**: cuenta los dígitos de un entero pasado como parámetro.
- **parseNumber**: transforma un número escrito en caracteres ASCII al double correspondiente.
- **isNumeric**: devuelve 1 si recibe un número y 0 si no.
- **clear**: limpia la pantalla.
- **strcmp**: compara dos strings.
- **startsWith**: verifica si el string en el primer parámetro comienza con el string pasado como segundo parámetro.

Problemas

Ante cualquier dificultad, nos pueden contactar en el correo institucional del ITBA a cualquiera de los integrantes del grupo.