California State University, Sacramento
College of Engineering and Computer Science

Computer Science 35: Introduction to Computer Architecture

CSC35.o Library 1.6 – x64 – Guide

How to Use

This library, but itself, is not a complete program. It must be linked into the object file that you will create during lab. For example, if you created the object file "lab.o", the following command will create a program called "a.out".

```
ld -o a.out csc35.o lab.o
```

Miscellaneous Subroutines

Subroutine	Input	Output	Notes
PrintAbout	*none*	*none*	Prints information about this library.
EndProgram	*none*	*none*	Terminates your program. You must call this subroutine at the end of your program.
PrintRegisters	*none*	*none*	Prints the contents of the register file to the screen.

String Subroutines

Subroutine	Input	Output	Notes
PrintCString	**rax**	*none*	Prints a null-terminated string stored in the address **%rax**.
ScanCString	**rax, rbx**	*none*	Scans a null-terminated string and stores it into the address **%rax**. The register **%rbx** must contain the maximum number of characters that can be read (the size of the buffer).
LengthCString	**rax**	**rax**	Returns the length of a null-terminated string stored at address **%rax**. The result is returned in **%rax**.
PrintChar	**al**	*none*	Prints the ASCII character stored in **%al** to the screen.
ScanChar	*none*	**al**	Scans an ASCII character from the keyboard and stores the result in **%al**. The rest of the **%rax** register is cleared.

Integer Subroutines

Subroutine	Input	Output	Notes
`PrintInt`	`rax`	*none*	Prints a signed integer stored in **%rax**.
`ScanInt`	*none*	`rax`	Scans a signed integer and stores it in **%rax**.
`PrintHex`	`rax`	*none*	Prints the integer, stored in **%rax**, to the screen in hexadecimal format.
`PrintHexByte`	`al`	*none*	Prints the byte, stored in **%al**, to the screen in hexadecimal format.
`Random`	`rax`	`rax`	Returns a random integer from 0 to (rax - 1) into **%rax**.

VT100 Subroutines

When you connect to another computer, often the Telnet software emulates a VT100 terminal screen. This standard supports color, screen formatting, and much more. The PuTTY software, which we are using for our labs, supports it.

Subroutine	Input	Output	Notes
`ClearScreen`	*none*	*none*	Clears the screen and moves the cursor to the top-left corner.
`SetCursor`	`rax, rbx`	*none*	Moves the cursor to column **%rax** and row **%rbx**. Indexing starts at 1 in the top-left corner.
`SetForeColor`	`rax`	*none*	Sets the text to the color specified in **%rax**. Please see the table below.
`SetBackColor`	`rax`	*none*	Sets the background to the color specified in **%rax**. Please see the table below.

VT100 Color Codes

Code	Color
`0`	Black
`1`	Red
`2`	Green
`3`	Yellow

Code	Color
`4`	Blue
`5`	Magenta
`6`	Cyan
`7`	White