Using the Assembler

- a.out format is not standard BSSSD format
- produces Mach-O (Mach object) file format

```
as [option] ... [file] ...
```

Options

```
-o
--
-f
-g
-v
-n
-l
-L
-V
-W
-dynamic
-static
```

Architecture Options

```
-arch
-force_cpusubtype_ALL
-arch_multiple
```

PowerPC-Specific Options

```
-no_ppc601
-static_branch_prediction_Y_bit
-static_branch_prediction_AT_bits
```

Tools

- /usr/lib/dyld -> the dynamic linker, used by kernel to load and bind a program at runtime
  - kernel loads dynamic linker into a new process and executes it
  - dynamic linker loads the program, all frameworks, and all libraries used by program

- static linker -> tool that combines object files into final executable
- cc
- c++
- as -> creates object files from assembly language code files (generally used by compiler driver)
- ld -> used to combine Mach-O files (by compiler driver and as standalone tool)
- libtool -> used to create static and dynamic libraries (supersedes ranlib)

analyzing mach-o files

- otool -> lists the contents of specific sections and segments within a Mach-O file
    - includes symbolic disassemblers for each supported CPU architecture and it knows how to format the contents of many common section types
- nm -> display contents of Mach-O file's symbol table
- size -> display size of various segments
- hexdumps

```
hexdump filename
hexdump -c filename

od -x filename
od -xc filename

xxd filename
xxd -r filename
```

- in vim
    - open file
    - `:%!xxd`
    - edit
    - `:%!xxd -r`
    - save
- in Xcode
    - open file
    - cmd+shift+j
    - right click filename
    - Open as -> Hex
- lipo -> used to analyze binaries that contain images for more than one architecture

- file -> shows type of a file
- pagestuff -> displays information on each logical page that compose an image, including names of sections and symbols contained in each page
- additional - c++filt
- linux - addr2line, readelf
- objdump -unwind-info executable -> display unwind info contained in executable