Lecture 7
Homework 1
Object Formats
Memory Maps  Linking and binding

* Memory

* file system

Q: Addresses significant?
A: No. Linux kernels will not map anything in the used range. Note: These are virtual addresses. Collisions with other student's not possible.
Object File Format

- Responsibilities
  → hold data

- Example
  - int buf[1000]
  - int init.data[2] = 01, 02, 03
  - const init const.data
  - char *xp = "test"

main()
  int local
  buf, init.data
  const data
  P, Q local, main

Mmap:
  - proc map
  - file /dev/null

Munmap:
  - to create an .exe

Exe
  - text
  - rodata
  - data

Mmap:
  - shared libraries
  - malloc

Linking merges these together and resolves issues:
* statically vs dynamically linked exec

- Types of Executables

- static:
  - text
  - rodata
  - data
  - bss

- dynamic:
  - text
  - rodata
  - data
  - bss
  - relocation info

  - problem: all exec info in executable file
  - problem: no way to separate library code to independent secion

  - some exec info elsewhere (and referred to by relocation info section)