Why Discussion?

- To help you understand course materials better
  - Review, special topic
- To deal with quiz, homework and program
- To answer your questions
- To give you a chance to increase your final score

Definitions of operating system

- Resource allocator
  - Memory, I/O, disk, network
- Control Program
  - Coordinate processes, handle
- Intermediary between application and hardware
  - Interface between user and computer, provides easier ways for user
- Kernel
  - One program running at all times on the computer
  - Notice: kernel is just the core of operating system
  - loaded into memory at the last stage of bootstrap program
- What you bought when purchasing 'Operating system'

Introduction

- Name:
- Major:
- Status: junior, senior....
- What do you expect from this course
- Anything else

Memory hierarchy

Review about Register

Example of execution process

IR: Instruction register, storing command
PC: Program counter, address for next command
AC: Accumulator, storing intermediate arithmetic and logic results.
Interrupt

- An interrupt is an asynchronous signal from hardware indicating the need for attention or a synchronous event in software indicating the need for a change in execution (from wiki)
- Hardware interrupt and software interrupt
  - Hardware: devices
  - Software: application, process

Precise interrupt

- The Program Counter (PC) is saved in a known place.
- All instructions before the one pointed to by the PC have fully executed.
- No instruction beyond the one pointed to by the PC has been executed.
- The execution state of the instruction pointed to by the PC is known.
  - Current state should be stored somewhere else before executing interrupt

Typical Interrupt

1. Device driver initiates I/O
2. CPU executes checks for interrupt between instructions
3. CPU receiving interrupt trigger a interrupt handler
4. Interrupt handler processes data returns from interrupt
5. CPU resumes processing of interrupted task

Execute of interrupt

1. Push the current process state into stack
2. Using the interrupt code to get the address of the interrupt handler
3. (Disable interrupt handler)
4. Execute handler
5. After finishing the interrupt handler, pop the process state to return control to former process

System call

- System call is caused by user application to request services from OS
- System call can be implemented by software interrupt
- A software interrupt is an interrupt generated within a processor by executing an instruction