Chapter 3: Processes

- Process Concept
- Concurrency
- Race Conditions
- Process Creation
- Interprocess Communication
- Examples of IPC Systems

What is a Process?

- Fundamental building block of modern operating systems is the notion of a *process*
- A process is a running program (a program in execution). This includes:

  Thread is execution context

  - All programs running on behalf of users (application programs)
  - Some operating system functions are also implemented using processes
- A process is a single thread of execution which is under the control of the CPU (this is different from a user thread)

Process Details

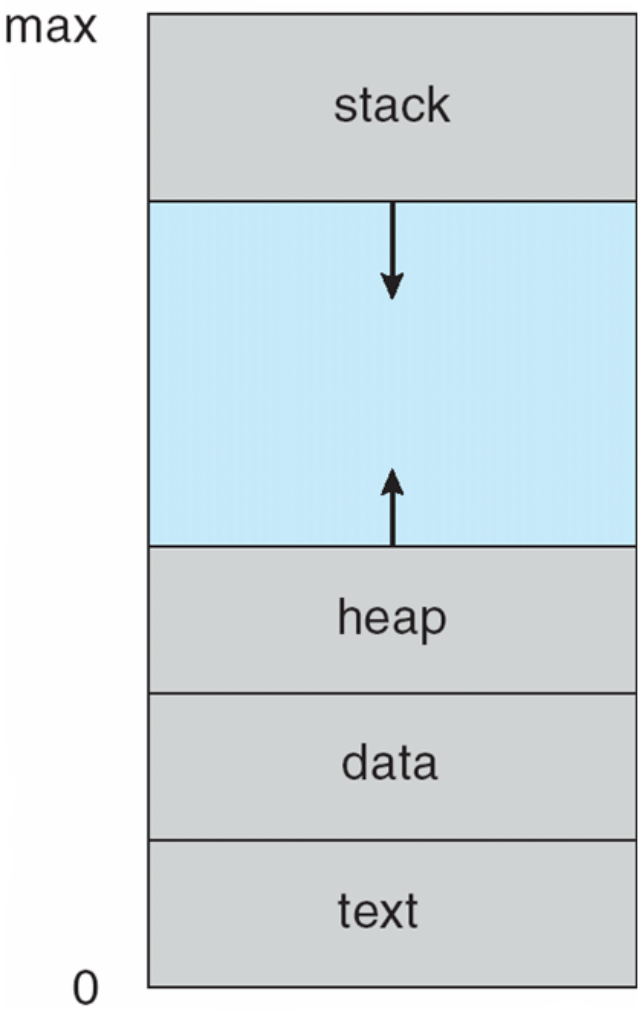
- Much of the functionality of a modern OS is the work required to manage processes
- OS may have hundreds of processes active at the same time
  - Although only a small number of them are executing in a multi-core system
- Processes are not found in the operating system *kernel*

Because the kernel has to manage processes

What is not a Process?

- A program by itself is not a process
- There is no one-to-one correspondence between programs and processes
  - E.g. there may be 10 people using emacs at the same time, i.e. 10 processes running emacs, but only one copy of the emacs program on disk
  - E.g. there may be many programs on disk that are not executing at the present time → not processes
  - Programs are passive entites, processes are active

A Process in Memory



- `text:` the instructions (actual machine code)
- `data:` the data that the program uses
- `heap:` used for dynamic memory
- `stack:` used for function calls

Process States

Modern OSes allow for more than one process to exist at the same time, and since there is usually only one processor, processes must assume different states during their lifetime:

- Running: currently being executed by the processor
- Blocked: waiting for some external event (e.g. I/O operation
- Ready: waiting for a turn at the processor
- Deadlocked: waiting for an event that will never happen

Process State Diagram

Process Control Block (PCB)

Information associated with each process

- Process state
- Program counter
- CPU registers
- CPU scheduling information
- Memory-management information
- Accounting information
- I/O status information

Process Control Block (PCB)