In virtual memory, a page holds words, the page gets put into a page frame in main memory, a page table holds the location of the pages that are relevant to that process

Logical address : address of words assigned in a process

Active processes are those that are currently in main memory

A child process becomes a copy of a parent by duplication the logical address page of the parent process. The child processes start as this and can load a new program into its address space

If pid is 0, it is a child process, which will then get the system call exec() to get a new program

The child can read from the same place as the parent, but write to a new place

Fork() system call makes a new process

IPCs:

* Shared Memory: both processes share memory and access it one at a time
  + Write to a file with one process and read the file with another
  + System call is only needed to make the shared memory
  + Issues with race conditions: final result depends on the random order the processes were called
* Message Passing: one process creates a message that is then delivered to the other process for reading
  + Allows for shared processes across different machines, unlike
  + Send and received message system calls are needed every time
  + Can be direct to the consumer or indirect with a mailbox
  + Can be either blocking or nonblocking
    - Blocking send blocks the sending process until the message is received
    - Nonblocking send allows the sending process to keep working after sending the message (commonly used)
    - Blocking receive blocks the receiver until a message arrives
    - Nonblocking receive allows the process to keep working after issuing a receive operation if there are no messages available (rare case)
* Piping:
  + Unnamed pipes can only communicate between parent and child
* Sockets:
  + Client process makes a request of the server
  + Sends messages between sockets over the internet
    - The socket sends a message through a port which then arrives at a port of a different machine and goes to a socket.
    - Socket address is a combination of the port number and IP address
    - Receiving port needs to be specified (use a port above 1024, since those below are commonly used for other things (aSocket = new DatagramSocket(6789);)), sending socket does not (aSocket = new DatagramSocket();)
  + RMI is only used when Java is being used, otherwise it’s RPC