資訊系統原理作業二補充說明

- fork() – create a child process
  - #include <sys/types.h>
  - #include <unistd.h>
  - pid_t fork(void)
- Returns:
  - 0 in child
  - process ID of child (>0) in parent
  - -1 on error

fork()

- A Simple Example:

```
…
  pid_t pid = fork();
  if (pid<0)               /* error */
  { …  }
  else if (pid == 0)       /* child */
  { …  }
  else                     /* parent */
  { …  }
…
```

Pipe (1/2)

- Pipes:
  - The oldest (and most commonly used) form of UNIX IPC (Inter-Process Communication)
  - half-duplex (data flows only in one direction)
  - Can be used only between processes that have a common ancestor

Pipe (2/2)

- pipe() – Create a pipe
  - #include <unistd.h>
  - int pipe(int fd[2]);
  - Returns: 0 if OK, -1 on error.
  - fd[0]: for reading, fd[1]: for writing
- Close unnecessary fds
- Read(fd[0])
- Write(fd[1])

References

- *Advanced Programming in the UNIX Environment*, by W. Richard Stevens, Addison-Wesley.
- Manual pages on UNIX systems:
  - man fork
  - man pipe