Operating Systems

Mediates between hardware + software.

- managing access to shared resources.
  (CPU, network, disk, RAM, printer, ...)
- providing services to make using resources easier. ← Project 12.
- enforce security + access policies.

Sys

init:
- call all other init functions.
- call Main.main
- call Sys.halt.

halt: infinite loop.

wait: loop a bunch.

error: print eg ERR6,
then call Sys.halt.

Array

new: call Memory.alloc.
dispose: Memory.deAlloc(this).

String

- field for max length.
- field for current length
- field for Array of characters.
- charAt, setCharAt, append, erase
  - all do index bounds checking.

intValue / setInt : Convert int ⟷ String.

String → int:

- loop as long as characters are digits.
  '0' = 48 .. '9' = 57.
  - keep track of current value
  - For each new digit,
    cur = 10 * cur + digit.
    ↑ subtract 48 from character.

int → String:

- to get last digit,: % 10
- chop off last digit : / 10.
- repeat loop until 0.

Mod!?

$$a \% d = a - (a/d) * d$$

how to deal w/ reverse order?

- reverse String.
- using recursion:
  - get the last digit
  - recursively convert n/10 to String
  - append the last digit.

Note: handle negatives!

Keyboard

keyPressed : just call Memory.peek

readChar : see pseudocode in book.

readLine :

- call readChar in a loop
- if backspace,
  - eraseLastChar from buffer
  - Output.backSpace.

Math

abs, max, min — easy.