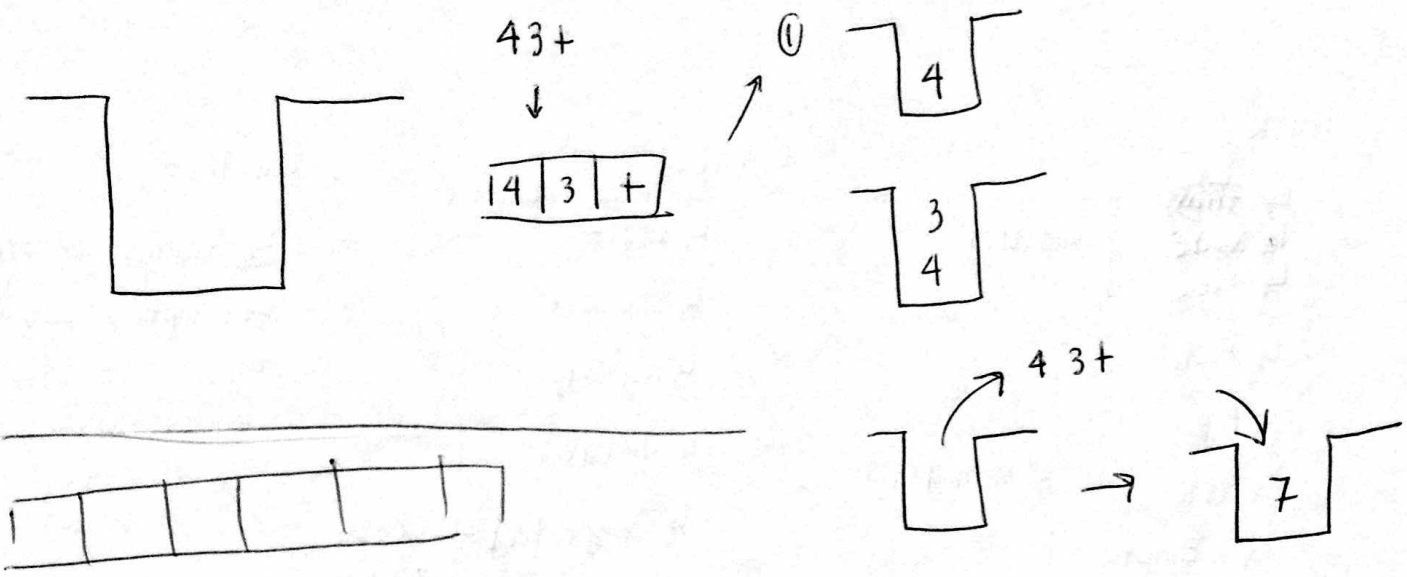
① Scan input → String

② Split String → iterate into array for strings ↑ → change to array of doubles

③ Move array into stack ← Run operations as it goes into stacks

example



if +

else if −

else if *

else if /

else { // would be #

Pang & Will

P
+
—
*
/

- Tester class : CalcTest. ~~class~~
- Calculator class : Calc ~~Cla~~
- Stack class : ~~Calc~~ Stack

- 1 Stack
  - ↳ Array
  - ↳ Size
  - } variables
  - ↳ Push
  - ↳ Pop
  - ↳ Peek
  - ↳ is Empty
  - ↳ size
  - } methods

2. Calc
   - ↳ Stack : variable
   - ↳ addit
   - ↳ subtract
   - ↳ multiply
   - ↳ divide
   - ~~↳ String total~~ — ~~variable~~ ≠ same as last # in stack
   - ↳ interpreter
   - } methods

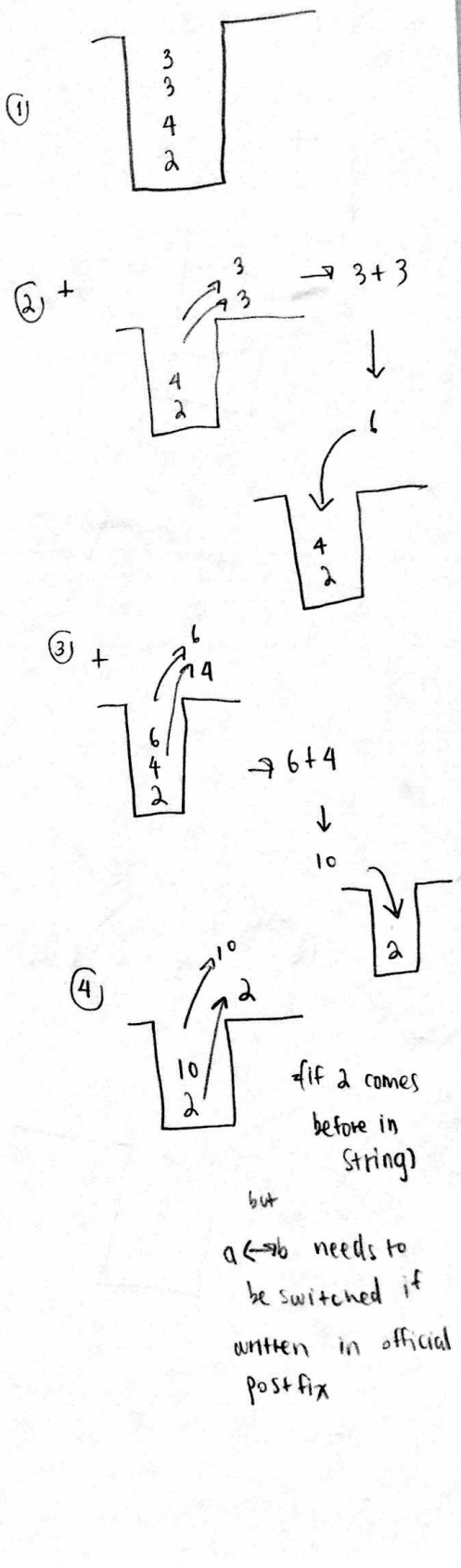
3. CalcTest
   - ↳ Scanner for I/O
   - ↳ Prompt. → methods
   - ↳ total = top (last thing in stack)

2 Calc

↳ interpreter

33 + 4 + 2 / or 2 4 3 3 + + / or 4 3 3 + + 2 /

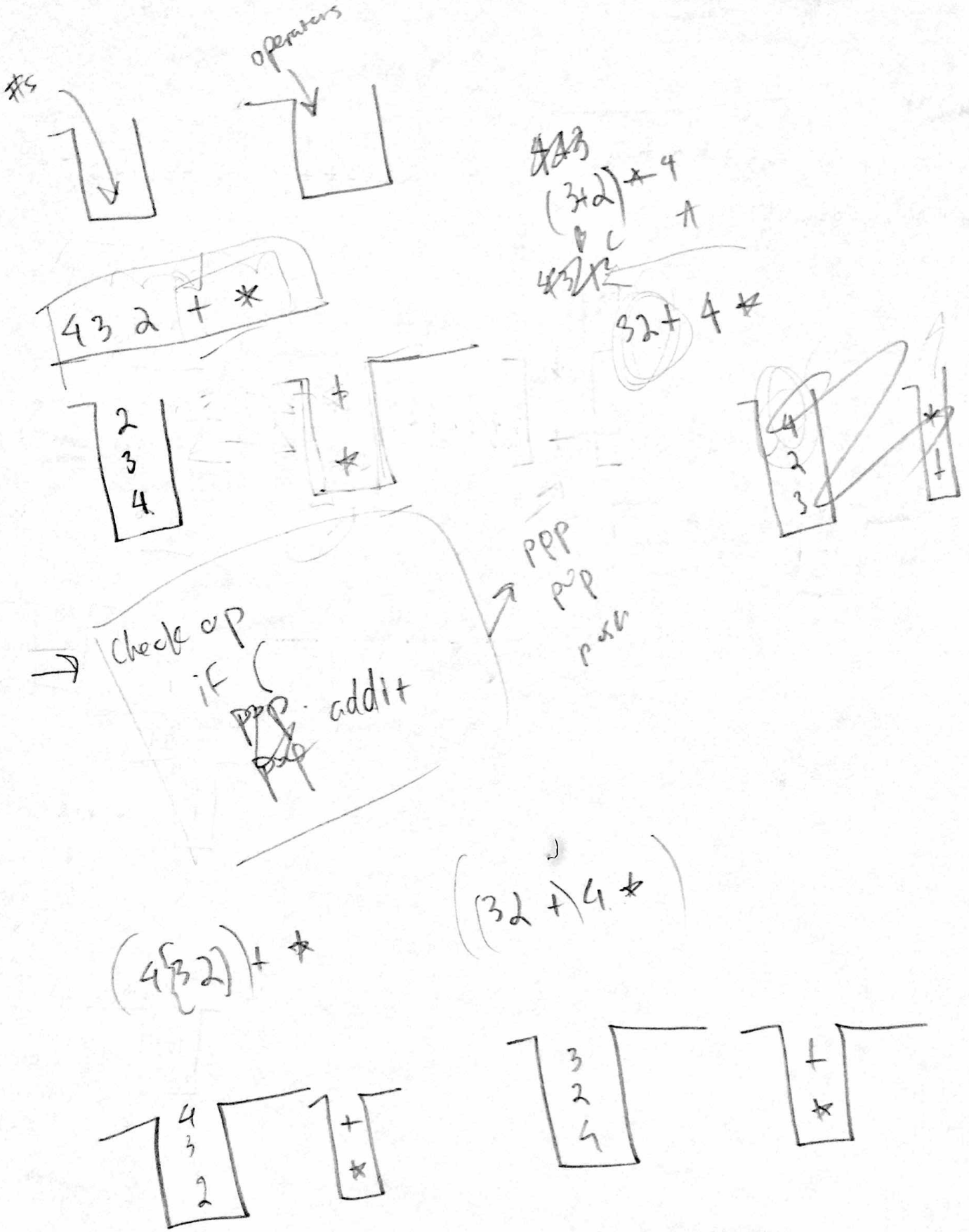
addIt (double a , double b)



*if 2 comes before in String)

but

a ⟷ b needs to be switched if written in official postfix


#s
operators
(3+2)*4
4 3 2 + *
3 2 + 4 *
2
3
4
+
*
4
2
3
*
1
Check op
if (
PPP . addIt
PPP
PPP
push
(4(3 2)) + *
(3 2 +) 4 *
4
3
2
+
*
3
2
4
1
*