exercise: calculator state

<u>DO NOT</u> look at succeeding pages yet

for a 4 function calculator

①   write a description of the state

②   define the effect of the following actions:

type_digit(d)	–	user presses single digit
equals	–	user presses '=' button
op(p)	–	user presses '+','–', '*' or '/' button

HINTS:

* interleave 1 & 2
  as you try to define operations you will find what you need to have in the state

* use scenarios to test your definitions of the operations
  run through the scenario writing down the operation that would happen and the new state at each step

calculator state – first attempt

total:	Nat	–	running total (accumulator)
disp:	Nat	–	number currently displayed
pend_op:	{+,–,*,/}	–	pending operation

pending operation needed when '=' is pressed)

actions – first attempt

type_digit(d):

> add d to the end of disp
> total and pend_op unchanged

equals:

> do pend_op to disp and total
> put result in both disp and total
> set pend_op to none

op(o):

> do pend_op to disp and total
> put result in both disp and total
> put o into pend_op

scenario

user types: 1 + 27 = – 3

start after 1 + 2

action	total	disp	pend_op
	1	2	+
type_digit(7)			
	1	27	+
equals			
	28	28	none
op(–)			
	28	28	–
type_digit(3)			
	28	283 !!!	–

calculator state – second attempt

total:	Nat	–	running total (accumulator)
disp:	Nat	–	number currently displayed
pend_op:	{+,–,*,/}	–	pending operation
typing:	Bool	–	flag

typing flag – user in the middle of typing a number

actions – second attempt

type_digit(d):

> if typing
>   then add **d** to the end of **disp**
>   otherwise clear **disp** and put **d** in it
>   also set **typing** to true
> **total** and **pend_op** unchanged

equals and op(o):
    as before except both

> set **typing** to false

scenario revisited

user types: 1 + 27 = – 3
start after 1 + 2

action	total	disp	pend_op	typing
	1	2	+	yes
type_digit(7)				
	1	27	+	yes
equals				
	28	28	none	no
op(–)				
	28	28	–	no
type_digit(3)				
	28	3	–	yes