Conversation with the customer:

Specifications

Functional Specifications: **collapse**

Abstraction: *collapse* collapses a number into a single digit that is the sum of all the digits, repeatedly applied.

method	description
collapse	

Invocation Prototype: collapse(value)

Parameters: *value* connotes an input number. It is a string of an integer value .GE. 0 consisting of between 1 and 50 digits. Mandatory. Arrives unvalidated.

Desired behavior (normal): *collapse r* epeatedly sums the individual digits of a value until the sum consists of only one digit. It returns the sum as a string. For example:

value	calculation	result
'0'	0	'0'
'10'	1 + 0 =	'1'
'123'	1 + 2 + 3 =	'6'
'99'	9 + 9 = 18 => 1 + 8 =	'9'
'98769'	9+8+7+6+9=39 => 3+9=12 => 1+2=	'3'

Desired behavior (errors):

If:	Any parameter violates its respective interface contract described above
Action:	Return Python's *None* built-in object.
Exit conditions:	No further processing takes place once an error is discovered.

State change: No state is retained or changed.

General examples:

Samples that illustrate behavior with normal input

Given:	collapse('0')	
Result:	'0'	
Given:	collapse('10')	
Result:	1'	
Given:	collapse('123')	
Result:	'6'	
Given:	collapse('99')	
Result:	'9'	
Given:	collapse('98769')	
Result:	'3'	
Given:	collapse(value='98769')	
Result:	'3'	

Samples that illustrate behavior with anomolous input

Given:	collapse()	
Result:	None	Rationale: No parameter was passed
Given:	collapse('a')	
Result:	None	Rationale: The input parm was not a string of an integer
Given:	collapse(1)	
Result:	None	Rationale: The input parm was not a string of an integer