

AP[®] COMPUTER SCIENCE A 2017 SCORING GUIDELINES

Question 1: Digits

Part (a)	Digits constructor	5 points
-----------------	--------------------	-----------------

Intent: *Initialize instance variable using passed parameter*

- +1 Constructs `digitList`
- +1 Identifies a digit in `num`
- +1 Adds at least one identified digit to a list
- +1 Adds all identified digits to a list (*must be in context of a loop*)
- +1 **On exit:** `digitList` contains all and only digits of `num` in the correct order

Part (b)	<code>isStrictlyIncreasing</code>	4 points
-----------------	-----------------------------------	-----------------

Intent: *Determine whether or not elements in `digitList` are in increasing order*

- +1 Compares at least one identified consecutive pair of `digitList` elements
- +1 Determines if a consecutive pair of `digitList` is out of order (*must be in context of a `digitList` traversal*)
- +1 Compares all necessary consecutive pairs of elements (*no bounds errors*)
- +1 Returns `true` iff all consecutive pairs of elements are in order; returns `false` otherwise

Question-Specific Penalties

- 2 (q) Uses confused identifier instead of `digitList`

AP[®] COMPUTER SCIENCE A 2017 SCORING GUIDELINES

Question 1: Scoring Notes

Part (a) Digits constructor			5 points
Points	Rubric Criteria	Responses earn the point if they ...	Responses will not earn the point if they ...
+1	Constructs <code>digitList</code>		<ul style="list-style-type: none"> initialize a local variable instead of <code>digitList</code> create an <code>ArrayList<int></code>
+1	Identifies a digit in <code>num</code>	<ul style="list-style-type: none"> identify one digit of <code>num</code> or a length one substring/character of the <code>String</code> representation of <code>num</code> 	<ul style="list-style-type: none"> treat <code>num</code> itself as a <code>String</code> convert <code>num</code> to a <code>String</code> incorrectly
+1	Adds at least one identified digit to a list	<ul style="list-style-type: none"> call <code>add</code> for some <code>ArrayList</code> using the previously identified digit, even if that digit was identified incorrectly 	<ul style="list-style-type: none"> add <code>String</code> or <code>char</code> to <code>digitList</code> without proper conversion to the correct type
+1	Adds all identified digits to a list (<i>must be in the context of a loop</i>)	<ul style="list-style-type: none"> call <code>add</code> for some <code>ArrayList</code> using previously identified digits, even if those digits were identified incorrectly 	<ul style="list-style-type: none"> identify only 1 digit
+1	On exit: <code>digitList</code> contains all and only digits of <code>num</code> in the correct order	<ul style="list-style-type: none"> add to <code>digitList</code> even if it is not instantiated properly 	<ul style="list-style-type: none"> obtain a list with the digits in reverse order omit one or more digits add extra digits mishandle edge case, e.g., 0 or 10 make a bounds error processing the <code>String</code> representation of <code>num</code>
Part (b) <code>isStrictlyIncreasing</code>			4 points
Points	Rubric Criteria	Responses earn the point if they ...	Responses will not earn the point if they ...
+1	Compares at least one identified consecutive pair of <code>digitList</code> elements	<ul style="list-style-type: none"> compare two consecutive <code>Integers</code> using <code>compareTo</code> explicitly convert two consecutive <code>Integers</code> to <code>ints</code> and compare those with <code>>=</code>, <code><=</code> etc. use auto-unboxing to convert two consecutive <code>Integers</code> to <code>ints</code> and compare those with <code>>=</code>, <code><=</code> etc. 	<ul style="list-style-type: none"> access <code>digitList</code> as an array or string fail to call <code>.get()</code> compare using <code>!></code>
+1	Determines if a consecutive pair of <code>digitList</code> is out of order (<i>must be in context of a <code>digitList</code> traversal</i>)	<ul style="list-style-type: none"> determine the correct relationship between the two compared consecutive elements, even if the syntax of the comparison is incorrect 	<ul style="list-style-type: none"> fail to consider the case where the two elements are equal for the false case
+1	Compares all necessary consecutive pairs of elements (<i>no bounds errors</i>)		<ul style="list-style-type: none"> return early
+1	Returns true iff all consecutive pairs of elements are in order; returns false otherwise	<ul style="list-style-type: none"> compare consecutive pairs for inequality, but fail to consider the case when two elements are equal 	<ul style="list-style-type: none"> return prematurely via <code>if (...)</code> return false; else return true;