

AP[®] COMPUTER SCIENCE A

2018 SCORING GUIDELINES

Question 3: Code Word Checker

Class: <code>CodeWordChecker</code>	9 points
--	-----------------

Intent: Define implementation of a class to determine if a string meets a set of criteria

- +1** Declares header: `public class CodeWordChecker implements StringChecker`
- +1** Declares all appropriate `private` instance variables
- +3** Constructors
 - +1** Declares headers: `public CodeWordChecker(int __, int __, String __)` and `public CodeWordChecker(String __)`
 - +1** Uses all parameters to initialize instance variables in 3-parameter constructor
 - +1** Uses parameter and default values to initialize instance variables in 1-parameter constructor
- +4** `isValid` method
 - +1** Declares header: `public boolean isValid(String __)`
 - +1** Checks for length between min and max inclusive
 - +1** Checks for unwanted string
 - +1** Returns `true` if length is between min and max and does not contain the unwanted string, `false` otherwise

AP[®] COMPUTER SCIENCE A

2018 SCORING GUIDELINES

Question 3: Scoring Notes

Class <code>CodeWordChecker</code> 9 points			
Points	Rubric Criteria	Responses earn the point if they...	Responses will not earn the point if they...
+1	Declares header: <code>public class CodeWordChecker implements StringChecker</code>	omit keyword <code>public</code>	- declare class <code>private</code> - declare class <code>static</code>
+1	Declares all appropriate <code>private</code> instance variables		- declare variables as <code>static</code> - omit keyword <code>private</code> - declare variables outside the class
+3 Constructors			
+1	Declares headers: <code>public CodeWordChecker (int __, int __, String __)</code> and <code>public CodeWordChecker (String __)</code>	omit keyword <code>public</code>	- declare method <code>static</code> - declare method <code>private</code>
+1	Uses all parameters to initialize instance variables in 3-parameter constructor		- fail to declare instance variables - initialize local variables instead of instance variables - assign variables to parameters
+1	Uses parameter and default values to initialize instance variables in 1-parameter constructor	initialize instance variables to default values when declared	- fail to declare instance variables - initialize local variables instead of instance variables - assign variables to parameters
+4 <code>isValid</code> method			
+1	Declares header: <code>public boolean isValid (String __)</code>		- fail to declare method <code>public</code> - declare method <code>static</code>
+1	Checks for length between min and max inclusive		- fail to use instance variables - fail to declare the method header
+1	Checks for unwanted string		- fail to use instance variables - fail to declare the method header
+1	Returns <code>true</code> if length is between min and max and does not contain the unwanted string, <code>false</code> otherwise	have incorrect checks for length and/or containment, but return the correct value based on those checks	- fail to declare the method header - fail to return in all cases - only check one substring location for containment

**AP[®] COMPUTER SCIENCE A
2018 SCORING GUIDELINES****Question 3: Code Word Checker**

```
public class CodeWordChecker implements StringChecker
{
    private int minLength;
    private int maxLength;
    private String notAllowed;

    public CodeWordChecker(int minLen, int maxLen, String symbol)
    {
        minLength = minLen;
        maxLength = maxLen;
        notAllowed = symbol;
    }

    public CodeWordChecker(String symbol)
    {
        minLength = 6;
        maxLength = 20;
        notAllowed = symbol;
    }

    public boolean isValid(String str)
    {
        return str.length() >= minLength && str.length() <= maxLength &&
            str.indexOf(notAllowed) == -1;
    }
}
```

These canonical solutions serve an expository role, depicting general approaches to solution. Each reflects only one instance from the infinite set of valid solutions. The solutions are presented in a coding style chosen to enhance readability and facilitate understanding.