| **(a)** | isWordChain |  |  |
| --- | --- | --- | --- |
|  | **Scoring Criteria** | **Decision Rules** |  |
| **1** | Accesses all adjacent pairs of wordList  elements (*no bounds errors*) | Responses **can** still earn the point even if they  • also access non-adjacent pairs of elements  • return early, as long as bounds and indices would otherwise support accessing all necessary pairs  Responses **will not** earn the point if they  • fail to access elements of wordList  correctly | **1 point** |
| **2** | Determines whether an element of the list contains a previous element of the list | Responses **can** still earn the point even if they  • make just one comparison  • fail to make the comparison in the context of a loop  • compare every element to the first element of the list  • access pairs of wordList elements incorrectly  Responses **will not** earn the point if they  • make an incorrect call to indexOf or use the indexOf return value incorrectly | **1 point** |
| **3** | Returns appropriate boolean in both cases (*algorithm*) | Responses **can** still earn the point even if they  • incorrectly identify whether an element of wordList contains the previous element  Responses **will not** earn the point if they  • return an incorrect value due to an early return  • fail to return true or false | **1 point** |
|  |  | **Total for part (a)** | **3 points** |

| **(b)** | createList |  |  |
| --- | --- | --- | --- |
|  | **Scoring Criteria** | **Decision Rules** |  |
| **4** | Declares and constructs an  ArrayList<String> | Responses **will not** earn the point if they  • fail to declare an ArrayList | **1 point** |
| **5** | Accesses all elements of wordList  (*no bounds errors*) | Responses **can** still earn the point even if they  • return early, as long as bounds and indices would otherwise support accessing all elements  Responses **will not** earn the point if they  • fail to access elements of wordList  correctly | **1 point** |
| **6** | Identifies strings that begin with target  (*in the context of an* if) | Responses **can** still earn the point even if they  • access elements of wordList  incorrectly  Responses **will not** earn the point if they  • call String methods incorrectly  • identify all strings that contain  target  • use the substring method without a guard against an element too short to contain target | **1 point** |
| **7** | Constructs a String that is a copy of an element of the list with the correct number of initial characters removed | Responses **can** still earn the point even if they  • make a copy of a wordList element that does not start with target or is not long enough to contain target | **1 point** |
| **8** | Adds to the constructed list at least one String based on an element of the original list | Responses **can** still earn the point even if they  • add an incorrectly constructed  String  • have not constructed a list Responses **will not** earn the point if they  • call add incorrectly | **1 point** |

| **9** | Returns list containing all and only identified and revised strings in the appropriate order (*algorithm*) | Responses **can** still earn the point even if they  • incorrectly identify strings beginning with target  • call add incorrectly  Responses **will not** earn the point if they  • add the original, unrevised element to the list to be returned  • modify wordList or any of its elements  • return an incorrect value due to an early return | **1 point** |
| --- | --- | --- | --- |
|  |  | **Total for part (b)** | **6 points** |
|  |  | **Total for question 3** | **9 points** |