Marcus Brooks

**Assignment 1 Test Cases and PseudoCode**

**Test Cases**

**isValidPassword**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case #** | **Input** | **Actual Input** | **Expected Output** | **Actual Output** | **Did the test pass?** |
| 1 | Mercy1 |  | true |  |  |
| 2 | Pika |  | Throw LengthException |  |  |
| 3 | Pikachu |  | Throw NoDigitException |  |  |
| 4 | pikachu1 |  | Throw NoUpperAlphaException |  |  |
| 5 | PIKACHU1 |  | Throw NoLowerAlphaException |  |  |
| 6 | Piiiikachu1 |  | Throw InvalidSequenceException |  |  |

**isWeakPassword**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case #** | **Input** | **Actual Input** | **Expected Output** | **Actual Output** | **Did the test pass?** |
| 1 | Pika1 |  | False |  |  |
| 2 | Pikachu1 |  | True |  |  |
| 3 | PikachuTheElectricMousePokemon1 |  | False |  |  |

**invalidPasswords**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case #** | **Input** | **Actual Input** | **Expected Output** | **Actual Output** | **Did the test pass?** |
| 1 | Pikachu1, Pikachu2,  Pikachu3, |  | Empty ArrayList |  |  |
| 2 | Pikachu1,  Pika,  Pikachu2 |  | Pika |  |  |
| 3 | Pikachu,  Pikachu1,  pikachu2 |  | Pikachu |  |  |
| 4 | pikachu1,  Pikachu2,  Pikachu3 |  | pikachu1 |  |  |
| 5 | Pikachu2,  Pikachu3,  PIKACHU1, |  | PIKACHU1 |  |  |
| 6 | Pikachu2,  Piiikachu1,  Pikachu3 |  | Piiikachu1 |  |  |

**PseudoCode**

**isValidPassword**

If passwordString is less than 6 digits in length

Throw LengthException

Else if passwordString does not contain a number

Throw NoDigitException

Else if passwordString does not contain an uppercase letter

Throw NoUpperAlphaException

Else if passwordString does not contain a lowercase letter

Throw NoLowerAlphaException

Else if passwordString contains more than 2 of the same character in a row

Throw InvalidSequenceException

Else

Return true

**isWeakPassword**

If passwordString is greater than or equal to 6 but less than or equal to 9

Return true

Else

Return false

**invalidPasswords**

Repeat for length of passwords list

Try

Call isValidPassword with an argument of the current password in the list

Catch LengthException

Add the invalid password and the error message to invalidPasswords list

Catch NoDigitException

Add the invalid password and the error message to invalidPasswords list

Catch NoUpperAlphaException

Add the invalid password and the error message to invalidPasswords list

Catch NoLowerAlphaException

Add the invalid password and the error message to invalidPasswords list

Catch InvalidSequenceException

Add the invalid password and the error message to invalidPasswords list

End loop

Return the invalidPasswords list