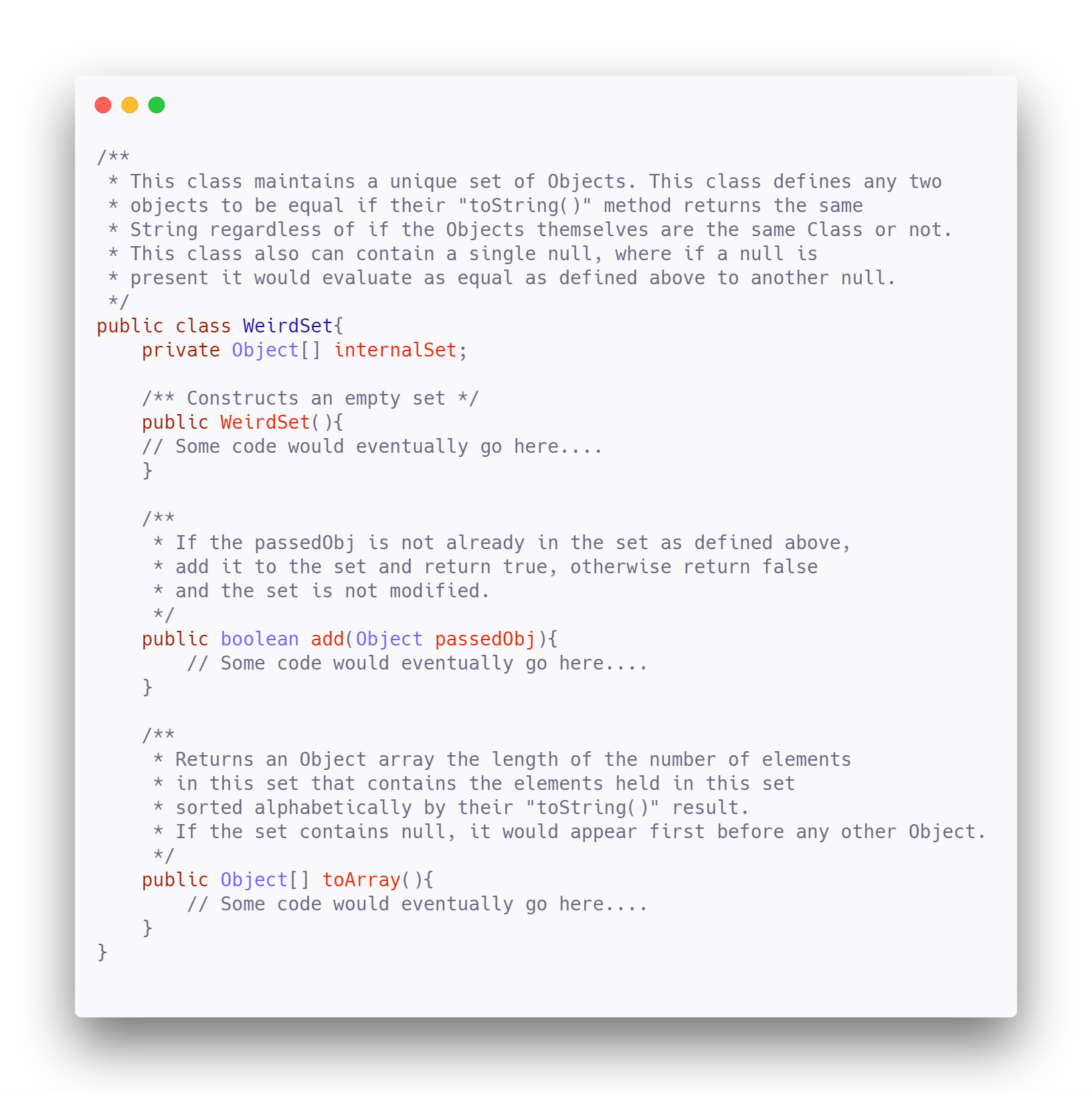
**Homework 3**

Chapter 6: Input Space Partitioning (ISP)

**Name: Antonio Zea Jr.**

**What to do?**

Using the interface description below for the **weirdSet** class, apply ISP and identify BCC (Base Choice Coverage) requirements. Be sure to clearly label your base choice. Your ISP analysis should be based on not needing to fully understand the inner workings of the code beyond what is described in the method comments. Using the code below complete the following 4 tables in the next page:



1. Using the code on the previous page complete the following 4 tables

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Table A: Identify input space and determine characteristics** | | | | | | | |
| **Method** | **Parameters** | **Returns** | **Values** | **Exception** | **Char ID** | **Characteristic** | **Covered by** |
| WeirdSet() | state-internalSet | WeirdSet |  |  |  |  | C1 |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| add() | passedObj | boolean | Object, null |  | C1  C2 | Passed object is already in the set  Null is passed |  |
|  | state-internalSet |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| toArray() | state-internalSet | Object[] |  |  | C3 | internalSet contains null and appears first |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table B: Design Partitioning** | | | | | | | | | | | |
| **Char ID** | **Characteristic** | | | **f1: WierdSet()** | **f2: add()** | | **f3: toArray()** | | **Partition** | | |
| C1  C2 | Passed object is already in the set  Null is passed | | | X | X  X | | X  X | | True/False  True/False | | |
| C3 | internalSet contains null and appears first | | |  |  | | X | | True/False | | |
| **Table C: Define test requirements for Base Choice Criteria (BCC) (make sure to indicate your base case)** | | | | | | | | | | |
| **Method** | | **Characteristics** | **Test Requirements** | | | **Infeasible TRs** | | **Revised TRs** | | **# TRs** |
| WierdSet() | | C1 | **T**, F | | |  | |  | | 2 |
| add() | | C1C2 | **TT**, TF, FT | | |  | |  | | 3 |
| toArray() | | C1C2C3 | **TTT**, TTF, TFT, FTT | | | TTF, TFT | | TTF->TFF, TFT->FFF | | 4 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case design** | | | |
| **Method** | **Case** | **Set up data** | **Steps** |
| WierdSet() |  | w = new WierdSet() | assertEquals(0, w.toArray().length) |
| Criteria: C1 |  |  |  |
|  |  |  |  |
|  |  |  |  |
| add() |  |  |  |
| Criteria:C1C2 | TT | base, w.add(null), w.add(null) | assertEquals(2, w.toArray().length) |
|  | TF | base, w.add(“mars”), w.add(“mars”) | assertEquals(2, w.toArray().length) |
|  | FT | base, w.add(“mars”), w.add(null) | assertEquals(3, w.toArray().length) |
| toArray()  Criteria:C1C2C3 | TTT | base, w.add(null), w.add(null) | assertEquals(“[“null”,“jupiter”]”, w.toArray().toString()) |
|  | TFF | base, w.add(“mars”), w.add(“mars”) | assertEquals(“[“jupiter”,“ mars”]”, w.toArray().toString()) |
|  | FFF | base, w.add(“earth”), w.add(“mars”) | assertEquals(“[“earth”,“jupiter”,“ mars”]”, w.toArray().toString()) |
|  | FTT | base, w.add(“earth”), w.add(“mars”),w.add(null) | assertEquals(“[“null”, “earth”,“jupiter”,“ mars”]”, w.toArray().toString()) |
|  |  |  |  |

|  |  |
| --- | --- |
| Base Data | w = new WierdSet(), w.add(“jupiter") |

**Due Date**

This homework is due by **Sunday, March 6, 2022, 11:59 pm.** A penalty of 10% per day will be deducted from your grade, starting at 12:00:01 am.

**What to submit?**

Submit the following file to Blackboard:

* A word document describing your answers to the questions above.