**­­Mark Kazzaz**

**Bellevue College – CS211**

**Assignment 4 – Chapter 11**

**2018-10-23**

|  |  |
| --- | --- |
| This is a resubmission of Chapter 11’s programming assignment. Previous errors in original submission have been resolved. Now my StableMarriage.main() handles both the short.dat file and the stable.dat file and meet’s all of the assignment’s requirements. Below I document screenshots of debugging in Intellij that helped determined my code was operating as intended. |  |

|  |  |
| --- | --- |
| **Requirement** | **Supporting screen** |
| Set each person to be free | I wrote a method that would set all Person objects in a given list to be free:  Code:  Debugging cycling through Person objects and seeing their resulted partner set to NOBODY after first setting all Person objects in list to be assigned to Person 0: |

|  |  |
| --- | --- |
| some man m with a nonempty preference list is free | I wrote a method to cycle through the first list passed a parameter to determine if there was still someone marked free with available choices:    As the program iterates, it’s using a while loop on stillOpen() to determine if there are still Person objects in list1 that (1) are free and (2) have spouse options available. When stillOpen() is true, it creates a needToAssign array of those eligible Person objects. Below are screenshots at the start and in the middle of execution that show the needToAssign array being reduces as list1 Person objects are assigned spouses. |

|  |  |
| --- | --- |
| w = first woman on m's list;  if (some man p is engaged to w) {  set p to be free  } | During execution, Woman 1 was found to be already assigned to Man 1 when processing Man 2. The requirement in this situation is to set Man 1 to be free. The following screenshots step through Man 1’s assignment to Woman 1 being removed. |

|  |  |
| --- | --- |
| set m and w to be engaged to each other | Easy enough transaction but the following screenshots show   1. Woman 1 being reassigned to Man 2 2. Man 2 being initially assigned to Woman 1   Before assignment:    After assignment: |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| for (each successor q of m on w's list) {  delete w from q's preference list  delete q from w's preference list  } | After the Woman has been assigned, I   1. build an array of who is less preferable of a spouse 2. use removalAll(array) to remove those less preferable matches from the Woman’s list 3. Loop through those less preferable matches to remove the related woman from their remaining choices   Code:    Array of less preferable matches as a Woman has been assigned a match:    Removal of less preferable matches from Woman’s list:    Remove of Woman 3 from the preferences of Man 1 & Man 2:   |  |  | | --- | --- | | Before | After | |  |  | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Results against stable.dat file | |  |  | | --- | --- | | Men | Women | |  |  | |