1. Problem Analysis

Officers are being paired for patrolling the city. Partners are assigned ahead of time and a program is needed to check whether the pairing is done correctly. The program should check whether each officer is paired with the same consistently in both lists, as well as any incorrect matches of self-assignment, and lastly output a result based on the inputs using the words “Good” or “Bad”.

|  |  |  |  |
| --- | --- | --- | --- |
| **Known Facts** | **User Requirements** | **Necessary Processing** | **Alternative Solutions** |
| -The number of officers  -One line will have name of officers  -Line after the names will have the names of who is paired with who.  -Output “good” if officers all have a partner that is not themselves  -Output “bad” if officer does not have partner, or they are partnered with themselves. | -Input number of officers (first line).  -Input names of officers (second line).  -Input who is paired with who (third line). | -Open text file with the used inputted information  Declare variables:  -Bool statement to determine outcome of pairings.  -int for number of officers  -2 string arrays for each list of names (number of indices for arrays from int of officers)  -Process the second and third lines and see if the pairings are valid.  -Using if/else statements to come up with an output for pairings:  -if odd number of officer’s output:  -“The current pairing of officers is invalid. There is an odd number of officers on duty.” with “Bad pairings!”  -if an officer is missing a partner output: “ There seems to be a problem, an officer seems to not be paired up.” with “Bad pairings!”  -if pairings are good output: “Good parings!” | -Ask user to input data and store in a text file.  -First line will be for number of officers  -Second and third line will be for names.  -Names will be stored in an array.  -Compare the inputted values that are stored in arrays (the pairs).  -Output saying “Good!” or “Bad!” depending on pairings. |

Problem Definition Table:

IPO Chart:

|  |  |  |
| --- | --- | --- |
| **Input** | **Processing** | **Output** |
| -First line number of officers on duty  -Second line names of officers separated by spaces.  -Third line will be who is paired with who. | -Array that will store the name of the inputted officers.  -Bool statement to output if the pairing are good or bad | Depending on what pairs inputted will output one of the three:  -“Good pairings!”  -“The current pairing of officers is invalid. There is an odd number of officers on duty.” with “Bad pairings!”  -“ There seems to be a problem, an officer seems to not be paired up.” with “Bad pairings!” |

Psuedo-Code:

Open text file with user inputted data

Declare the variables:

Boolean statement (isGood) set initially to true for good pairings

Int that holds number of officers

nameList1 Array for first set of names

nameList2 Array for second list of names

Reads first line of text file (string value)

Convert the string into int value and that will be number of officers

if number of officers == even

for loop that splits line 2 and assigns split value to indices in an array up to last value

read last value of string separately

for loop that splits line 3 and assigns split value to indices in an array up to last value

read last value of string separately

for loop compares the two arrays

if(array nameList1 == array nameList2)

set isGood to false

output cannot pair officers with themselves

else

if(array nameList1 != array nameList2[indexNameList2])

set isGood to false

output officer missing partner

if isGood == false

output bad

else

output good

else

output odd number of officers on duty

Testing:

**-Test case 1 (Even number of officers and each officer is paired up with a partner that is not themselves.)**

Input:

6

Octopus Columbia Coconut Zumba Hummus Nerlans

Nerlans Hummus Zumba Coconut Columbia Octopus

Output:

Good pairings!

**-Test case 2 (Max number of consistent pairs. Works for values greater than 1 and less than 30. Tested with 30.)**

Input:

30

a b c d e f g h i j k l m n o p q r s t u v w x y z tyrone samosa vaseline tissue

b a d c f e h g j i l k n m p o r q t s v u x w z y samosa tyrone tissue vaseline

Output:

Good pairings!

**-Test case 3 (Officer paired up with themselves)**

Input:

6

Octopus Columbia Coconut Zumba Hummus Nerlans

Octopus Hummus Zumba Coconut Columbia Nerlans

Output:

An officer cannot be partnered up with themselves.

Bad pairings!

**-Test case 4 (Odd number of officers)**

Input:

5

Columbia Coconut Zumba Hummus Nerlans

Hummus Zumba Coconut Columbia Nerlans

Output:

The current pairing of officers is invalid. There is an odd number of officers on duty.

**-Test case 5 (Officer missing a partner)**

Input:

6

Octopus Columbia Coconut Zumba Hummus Nerlans

Nerlans Hummus Zumba Coconut Columbia

Output:

There seems to be a problem, an officer seems to not be paired up.

Bad pairings!