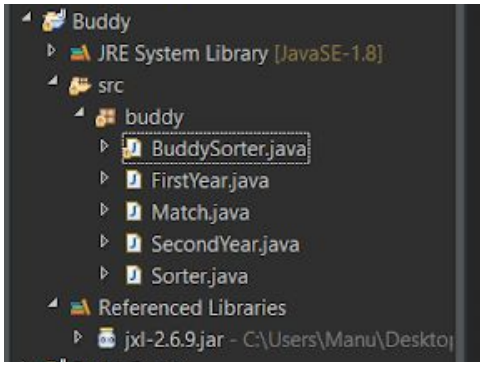
Classes:



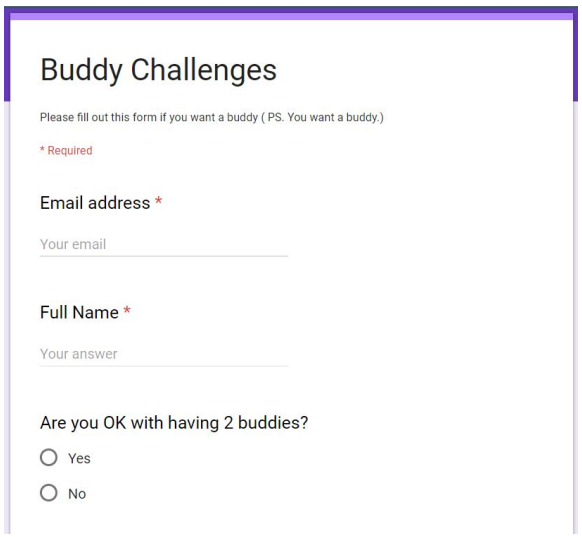
Program file

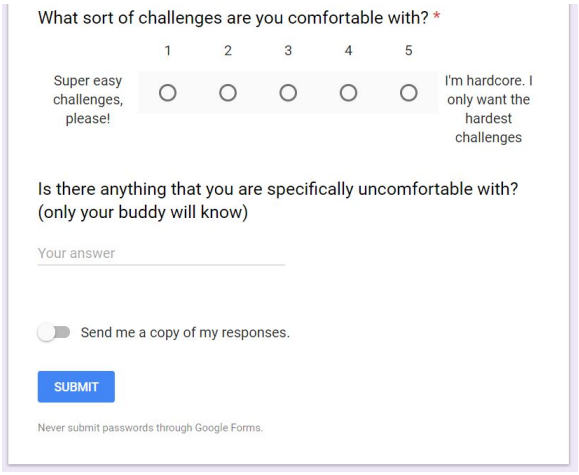
INSERT IMAGE OF FILE

Techniques Used:

Front End:

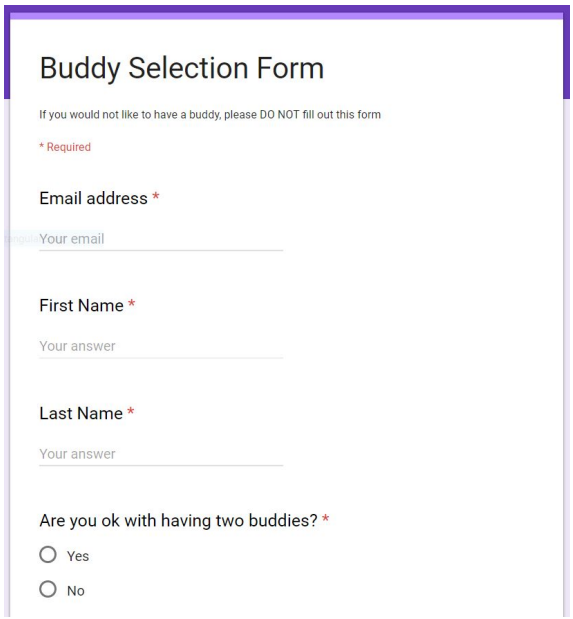
Create Google Forms for First Years and :

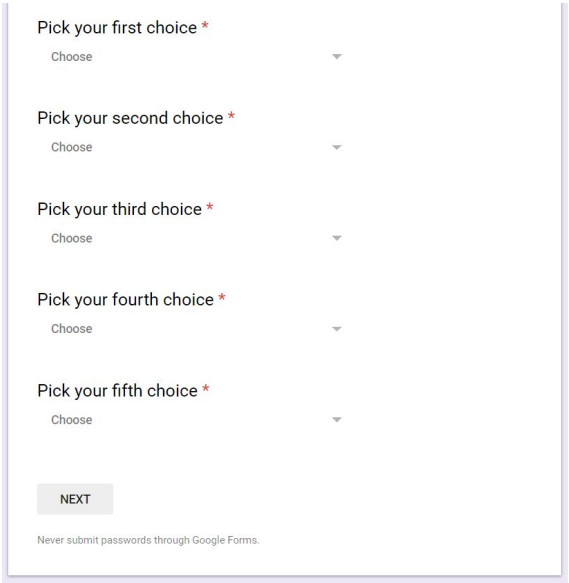


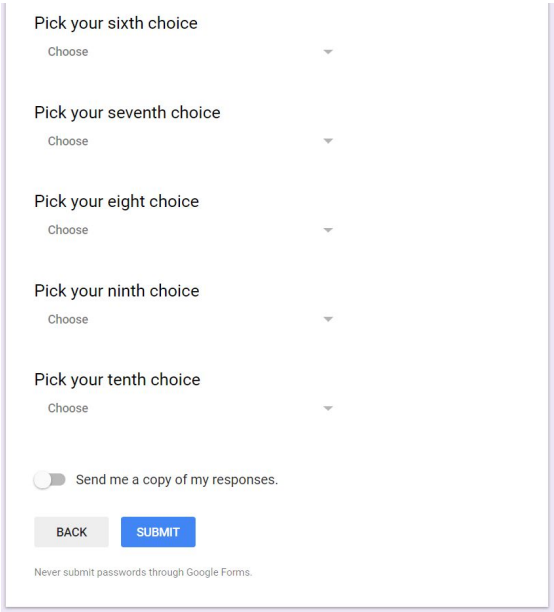


Second Years:

Create Google Forms for second years:

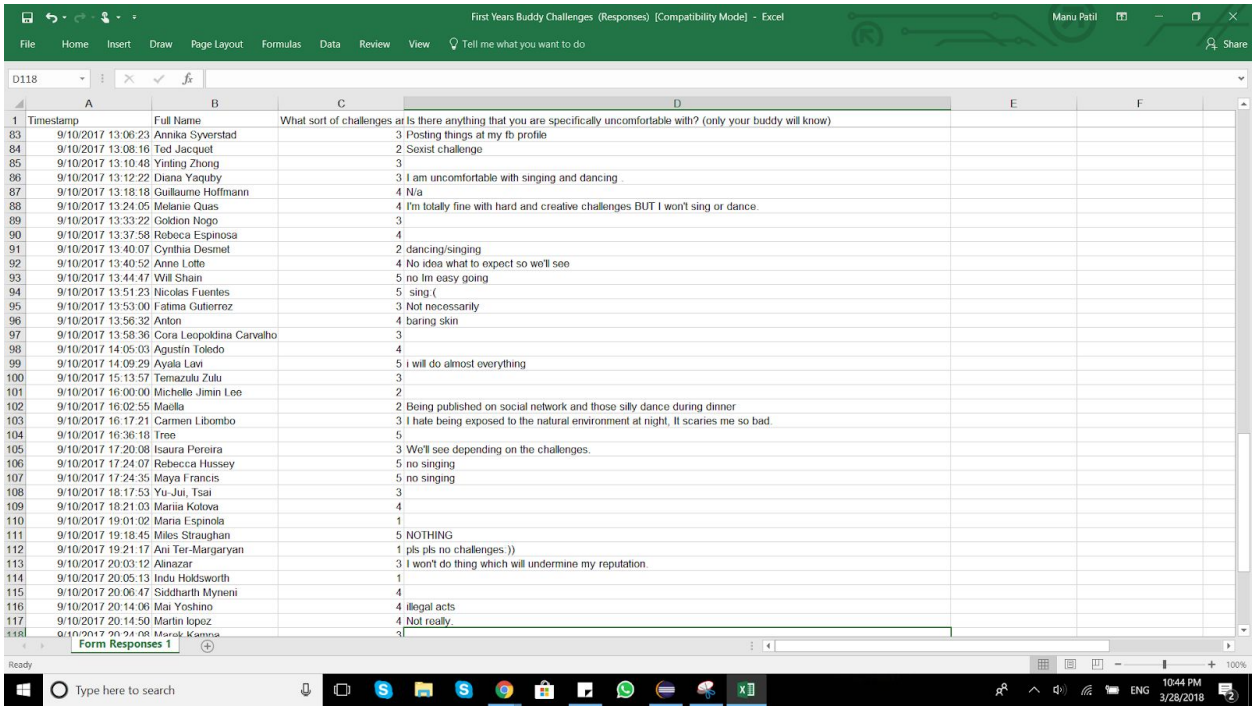




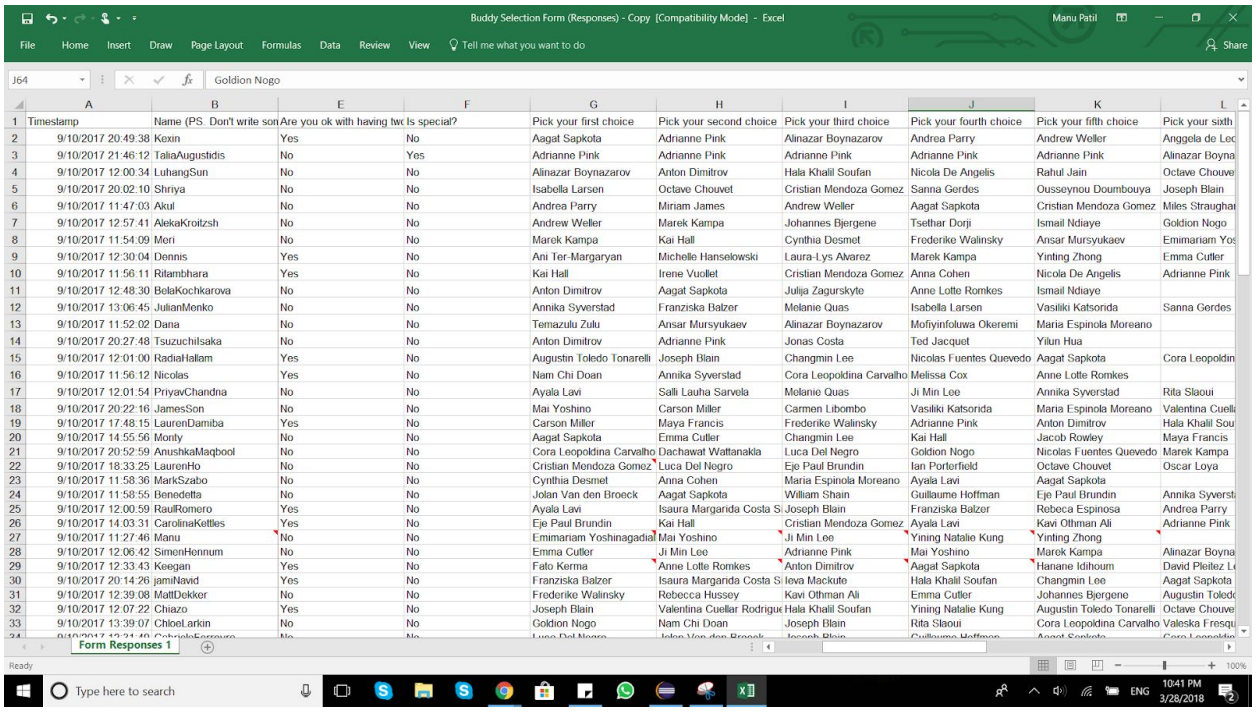


This technique was used as Google Forms provided an easy way to collect data from a large number of people. It was beyond my ability to create a database that would collect response from multiple users on multiple different machines. Additionally, Google Forms is a well tested program available for public use. The use of google forms allowed for users to update their answers within a time period with relative ease. Lastly, this provided a user interface that most students at this school were very familiar with which would serve to better the user experience

Create excel file from responses:



	Timestamp	Full Name	What sort of challenges ar	Is there anything that you are specifically uncomfortable with? (only your buddy will know)
83	9/10/2017 13:06:23	Annika Syverstad	3	Posting things at my fb profile
84	9/10/2017 13:08:16	Ted Jacquet	2	Sexist challenge
85	9/10/2017 13:10:48	Yinting Zhong	3	
86	9/10/2017 13:12:22	Diana Yaquby	3	I am uncomfortable with singing and dancing.
87	9/10/2017 13:18:18	Guillaume Hoffmann	4	N/a
88	9/10/2017 13:24:05	Melanie Quas	4	I'm totally fine with hard and creative challenges BUT I won't sing or dance.
89	9/10/2017 13:33:22	Goldion Nogo	3	
90	9/10/2017 13:37:58	Rebeca Espinosa	4	
91	9/10/2017 13:40:07	Cynthia Desmet	2	dancing/singing
92	9/10/2017 13:40:52	Anne Lotte	4	No idea what to expect so we'll see
93	9/10/2017 13:44:47	Will Shain	5	no Im easy going
94	9/10/2017 13:51:23	Nicolas Fuentes	5	sing:(
95	9/10/2017 13:53:00	Fatima Gutierrez	3	Not necessarily
96	9/10/2017 13:56:32	Anton	4	baring skin
97	9/10/2017 13:58:36	Cora Leopoldina Carvalho	3	
98	9/10/2017 14:05:03	Agustin Toledo	4	
99	9/10/2017 14:09:29	Ayala Lavi	5	i will do almost everything
100	9/10/2017 15:13:57	Temazulu Zulu	3	
101	9/10/2017 16:00:00	Michelle Jimin Lee	2	
102	9/10/2017 16:02:55	Maëlla	2	Being published on social network and those silly dance during dinner
103	9/10/2017 16:17:21	Carmen Libombo	3	I hate being exposed to the natural environment at night, It scaries me so bad.
104	9/10/2017 16:36:18	Tree	5	
105	9/10/2017 17:20:08	Isaura Pereira	3	We'll see depending on the challenges.
106	9/10/2017 17:24:07	Rebecca Hussey	5	no singing
107	9/10/2017 17:24:35	Maya Francis	5	no singing
108	9/10/2017 18:17:53	Yu-Jui, Tsai	3	
109	9/10/2017 18:21:03	Mariia Kotova	4	
110	9/10/2017 19:01:02	Maria Espinola	1	
111	9/10/2017 19:18:45	Miles Straughan	5	NOTHING
112	9/10/2017 19:21:17	Ani Ter-Margaryan	1	pls pls no challenges:))
113	9/10/2017 20:03:12	Alinazar	3	I won't do thing which will undermine my reputation.
114	9/10/2017 20:05:13	Indu Holdsworth	1	
115	9/10/2017 20:06:47	Siddharth Myneni	4	
116	9/10/2017 20:14:06	Mai Yoshino	4	illegal acts
117	9/10/2017 20:14:50	Martin lopez	4	Not really.
118	9/10/2017 20:24:08	Marek Kampa	3	



	Timestamp	Name (PS. Don't write son	Are you ok with having tw	Is special?	Pick your first choice	Pick your second choice	Pick your third choice	Pick your fourth choice	Pick your fifth choice	Pick your sixth
2	9/10/2017 20:49:38	Kexin	Yes	No	Aagat Sapkota	Adrianne Pink	Alinazar Boynazarov	Andrea Parry	Andrew Weller	Anggela de Le
3	9/10/2017 21:46:12	TaliaAugustidis	No	Yes	Adrianne Pink	Adrianne Pink	Adrianne Pink	Adrianne Pink	Adrianne Pink	Alinazar Boyna
4	9/10/2017 12:00:34	LuhangSun	No	No	Alinazar Boynazarov	Anton Dimitrov	Hala Khalil Soufan	Nicola De Angelis	Rahul Jain	Octave Chouve
5	9/10/2017 20:02:10	Shriya	No	No	Isabella Larsen	Octave Chouvet	Cristian Mendoza Gomez	Sanna Gerdes	Ousseynou Doumbouya	Joseph Blain
6	9/10/2017 11:47:03	Akul	No	No	Andrea Parry	Miriam James	Andrew Weller	Aagat Sapkota	Cristian Mendoza Gomez	Miles Straughar
7	9/10/2017 12:57:41	AlekaKroitzsh	No	No	Andrew Weller	Marek Kampa	Johannes Bjergene	Tsethar Dorji	Ismail Ndiaye	Goldion Nogo
8	9/10/2017 11:54:09	Meri	No	No	Marek Kampa	Kai Hall	Cynthia Desmet	Frederike Walinsky	Ansar Mursyukaev	Emimariam Yos
9	9/10/2017 12:30:04	Dennis	Yes	No	Ani Ter-Margaryan	Michelle Hanselowski	Laura-Lys Alvarez	Marek Kampa	Yinting Zhong	Emma Cutler
10	9/10/2017 11:56:11	Ritambhara	Yes	No	Kai Hall	Irene Vuollet	Cristian Mendoza Gomez	Anna Cohen	Nicola De Angelis	Adrianne Pink
11	9/10/2017 12:48:30	BelaKochkarova	No	No	Anton Dimitrov	Aagat Sapkota	Julija Zagurskyte	Anne Lotte Romkes	Ismail Ndiaye	
12	9/10/2017 13:06:45	JulianMenko	No	No	Annika Syverstad	Franziska Balzer	Melanie Quas	Isabella Larsen	Vasiliki Katsorida	Sanna Gerdes
13	9/10/2017 11:52:02	Dana	No	No	Temazulu Zulu	Ansar Mursyukaev	Alinazar Boynazarov	Mofiyinfoluwa Okeremi	Maria Espinola Moreano	
14	9/10/2017 20:27:48	Tsuzuchilsaka	No	No	Anton Dimitrov	Adrianne Pink	Jonas Costa	Ted Jacquet	Yilun Hua	
15	9/10/2017 12:01:00	RadiaHallam	Yes	No	Augustin Toledo Tonarelli	Joseph Blain	Changmin Lee	Nicolas Fuentes Quevedo	Aagat Sapkota	Cora Leopoldin
16	9/10/2017 11:56:12	Nicolas	Yes	No	Nam Chi Doan	Annika Syverstad	Cora Leopoldina Carvalho	Melissa Cox	Anne Lotte Romkes	
17	9/10/2017 12:01:54	PriyavChandna	No	No	Ayala Lavi	Salli Lauha Sarvela	Melanie Quas	Ji Min Lee	Annika Syverstad	Rita Slaoui
18	9/10/2017 20:22:16	JamesSon	No	No	Mai Yoshino	Carson Miller	Carmen Libombo	Vasiliki Katsorida	Maria Espinola Moreano	Valentina Cuell
19	9/10/2017 17:48:15	LaurenDamiba	Yes	No	Carson Miller	Maya Francis	Frederike Walinsky	Adrianne Pink	Anton Dimitrov	Hala Khalil Sou
20	9/10/2017 14:55:56	Monty	No	No	Aagat Sapkota	Emma Cutler	Changmin Lee	Kai Hall	Jacob Rowley	Maya Francis
21	9/10/2017 20:52:59	AnushkaMaqbool	No	No	Cora Leopoldina Carvalho	Dachawat Wattanakla	Luca Del Negro	Goldion Nogo	Nicolas Fuentes Quevedo	Marek Kampa
22	9/10/2017 18:33:25	LaurenHo	No	No	Cristian Mendoza Gomez	Luca Del Negro	Eje Paul Brundin	Ian Porterfield	Octave Chouvet	Oscar Loya
23	9/10/2017 11:58:36	MarkSzabo	No	No	Cynthia Desmet	Anna Cohen	Maria Espinola Moreano	Ayala Lavi	Aagat Sapkota	
24	9/10/2017 11:58:55	Benedetta	No	No	Jolan Van den Broeck	Aagat Sapkota	William Shain	Guillaume Hoffman	Eje Paul Brundin	Annika Syverst
25	9/10/2017 12:00:59	RaulRomero	Yes	No	Ayala Lavi	Isaura Margarida Costa Si	Joseph Blain	Franziska Balzer	Rebeca Espinosa	Andrea Parry
26	9/10/2017 14:03:31	CarolinaKettles	Yes	No	Eje Paul Brundin	Kai Hall	Cristian Mendoza Gomez	Ayala Lavi	Kavi Othman Ali	Adrianne Pink
27	9/10/2017 11:27:46	Manu	No	No	Emimariam Yoshinagadial	Mai Yoshino	Ji Min Lee	Yining Natalie Kung	Yinting Zhong	
28	9/10/2017 12:06:42	SimonHennum	No	No	Emma Cutler	Ji Min Lee	Adrianne Pink	Mai Yoshino	Marek Kampa	Alinazar Boyna
29	9/10/2017 12:33:43	Keegan	Yes	No	Fato Kerma	Anne Lotte Romkes	Anton Dimitrov	Aagat Sapkota	Hanane Idihoum	David Pleitez L
30	9/10/2017 20:14:26	jamiNavid	Yes	No	Franziska Balzer	Isaura Margarida Costa Si	Ieva Mackute	Hala Khalil Soufan	Changmin Lee	Aagat Sapkota
31	9/10/2017 12:39:08	MattDekker	No	No	Frederike Walinsky	Rebecca Hussey	Kavi Othman Ali	Emma Cutler	Johannes Bjergene	Augustin Toled
32	9/10/2017 12:07:22	Chiazo	Yes	No	Joseph Blain	Valentina Cuellar Rodrigue	Hala Khalil Soufan	Yining Natalie Kung	Augustin Toledo Tonarelli	Octave Chouve
33	9/10/2017 13:39:07	ChloeLarkin	No	No	Goldion Nogo	Nam Chi Doan	Joseph Blain	Rita Slaoui	Cora Leopoldina Carvalho	Valeska Fresq

Excel provided an interface to process large amounts of data. Additionally, this interface allows for easy distribution of data through a mail merge when the final output is produced.

Code for reading excel data into program

```java
public static final String FILE_LOCATION_1 = "C:\\Users\\Manu\\Desktop\\First Years Buddy Challenges  (Responses).xls";
public static final String FILE_LOCATION_2 = "C:\\Users\\Manu\\Desktop\\Buddy Selection Form (Responses).xls";
public static final String FILE_LOCATION_3 = "C:\\Users\\Manu\\Desktop\\output.xls";

private static ArrayList<FirstYear> firstYears;
private static ArrayList<SecondYear> secondYears;
private static ArrayList<Match> matches;

public static void main(String[] args) throws IOException, BiffException, RowsExceededException, WriteException {
    try {
        firstYearInput = Workbook.getWorkbook(new File(FILE_LOCATION_1));
        Sheet firstSheet = firstYearInput.getSheet(0);
        firstYears = new ArrayList<FirstYear>();
        for(int i = 1; i < firstSheet.getRows(); i++) {
            System.out.println(firstSheet.getCell(2, i).getContents());
            firstYears.add(new FirstYear(firstSheet.getCell(2, i).getContents(), "X"));

        }
```

This code allows for the access of data from the excel file. This was necessary as the alternative was to copy and paste data into files. This method was explored in a previous iteration and proved to complicate the program unnecessarily. The code for this attempt is shown below. The solution above is far more appropriate and concise.

```java
private static void getData() throws FileNotFoundException {
    in = new Scanner(new FileReader("FirstYears"));
    while (in.hasNextLine()) {
        String name = in.nextLine();
        firstYears.add(new FirstYear(name));
    }
    in.close();

    in = new Scanner(new FileReader("SecondYears"));
    Scanner inT = new Scanner(new FileReader("Two"));
    Scanner inS = new Scanner(new FileReader("Special"));
    Scanner in1 = new Scanner(new FileReader("Pick01"));
    Scanner in2 = new Scanner(new FileReader("Pick02"));
    Scanner in3 = new Scanner(new FileReader("Pick03"));
    Scanner in4 = new Scanner(new FileReader("Pick04"));
    Scanner in5 = new Scanner(new FileReader("Pick05"));
    Scanner in6 = new Scanner(new FileReader("Pick6"));
    Scanner in7 = new Scanner(new FileReader("Pick7"));
    Scanner in8 = new Scanner(new FileReader("Pick8"));
    Scanner in9 = new Scanner(new FileReader("Pick9"));
    Scanner in10 = new Scanner(new FileReader("Pick10"));
    while (in.hasNextLine()) {
        String name = in.nextLine();

        boolean isOk = (inT.nextLine().equalsIgnoreCase("Yes"));

        boolean isSpecial = (inS.nextLine().equalsIgnoreCase("Yes"));

        ArrayList<FirstYear> temp = new ArrayList<FirstYear>();
        temp.add(findFirsty(in1.nextLine()));
        temp.add(findFirsty(in2.nextLine()));
        temp.add(findFirsty(in3.nextLine()));
        temp.add(findFirsty(in4.nextLine()));
        temp.add(findFirsty(in5.nextLine()));
        temp.add(findFirsty(in6.nextLine()));
        temp.add(findFirsty(in7.nextLine()));
        temp.add(findFirsty(in8.nextLine()));
        temp.add(findFirsty(in9.nextLine()));
        temp.add(findFirsty(in10.nextLine()));

        secondYears.add(new SecondYear(name, temp, isOk, isSpecial));
        // System.out.println(name+ temp);
    }
    in.close();
    inT.close();
    inS.close();
    in1.close();
    in2.close();
    in3.close();
    in4.close();
    in5.close();
    in6.close();
    in7.close();
    in8.close();
    in9.close();
    in10.close();
    System.out.println("closed");
}
```

Code for writing data from program to excel:

The original solution of data input and output did not consider outputting the data outside of the program. A list was simply generated in the program and presented to the client. In a follow of meeting, my client suggested that the output be displayed in an excel file so that the data could be easily set up for a mail merge. The following code accomplishes this very task.

```
WritableWorkbook ww ;
ww = Workbook.createWorkbook(new File(FILE_LOCATION_3));
WritableSheet outSheet = ww.createSheet("Buddies", 1);
for (int i = 0; i < secondYears.size(); i ++) {
    outSheet.addCell(new Label(0,i,matches.get(i).getSecondYear().getName()));
    outSheet.addCell(new Label(1,i,matches.get(i).getFirstYear().getName()));
    outSheet.addCell(new Label(0,i,secondYears.get(i).getName()));
    outSheet.addCell(new Label(1,i,firstYears.get(i).getName()));

}
 ww.write();
 ww.close();
```

Back End:

The following three sections deal with how the data is stored before and after it is processed. The FirstYear and SecondYear classes hold the data for first years and second years, respectively. They hold the appropriate data such as name, region

Creating Class to represent First Year:

FirstYear.java

```java
package buddy;

public class FirstYear {
    private String name;
    private String region;
    private boolean isOkwith2 ;
    /**
     * @param name
     * @param secondName
     * @param region
     */
    public FirstYear(String name, String region, boolean isOkwith2) {
        super();
        this.name = name;
        this.region = region;
        this.isOkwith2 = isOkwith2;
    }
    public FirstYear() {
        super();
        this.name = "";
        this.region = "";
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }

    public String getRegion() {
        return region;
    }
    public void setRegion(String region) {
        this.region = region;
    }
    public boolean isOkwith2() {
        return isOkwith2;
    }
    public void setOkwith2(boolean isOkwith2) {
        this.isOkwith2 = isOkwith2;
    }
    @Override
    public String toString() {
        return "FirstYear [name=" + name + ", region=" + region + ", isOkwith2=" + isOkwith2 +
"]";
    }
```

Creating Class to represent Second Year:

SecondYear.java

```java
package buddy;

import java.util.ArrayList;

public class SecondYear {
    private String name;
    private ArrayList<FirstYear> selections = new ArrayList<FirstYear>();
    private String region;
    private boolean isOkwith2;
    /**
     * @param name
     * @param selections
     * @param region
     * @param isOkwith2
     */
    public SecondYear(String name, ArrayList<FirstYear> selections, String region, boolean isOkwith2) {
        super();
        this.name = name;
        this.selections = selections;
        this.region = region;
        this.isOkwith2 = isOkwith2;
    }
    public SecondYear() {
        super();
        this.name = null;
        this.selections = null;
        this.region = null;
    }

    public String getName() {
        return name;
    }
    public ArrayList<FirstYear> getSelections() {
        return selections;
    }
    public String getRegion() {
        return region;
    }

    public boolean isOkwith2() {
        return isOkwith2;
    }
    @Override
    public String toString() {
        String output = "SecondYear [name=" + name + ", region=" + region + ", isOkwith2=" + isOkwith2 + ", selections=" ;
        for int i = 0; i < selections.size(); i++) {
            output += selections.toString() + ", ";
        }

        return output.substring(0,output.length()-2) + "]\n";
    }


}
```

Creating Class to represent a Match:

Match.java

```java
package buddy;

public class Match {
    private SecondYear secondYear;
    private SecondYear secondSecondYear;
    private FirstYear firstYear;
    private FirstYear secondFirstYear;
    private boolean hasSecondSecond;
    private boolean hasSecondFirst;
    /**
     * @param secondYear
     * @param firstYear
     */
    public Match(SecondYear secondYear, FirstYear firstYear) {
        this.secondYear = secondYear;
        this.firstYear = firstYear;
        secondSecondYear = null;
        secondFirstYear = null;
        hasSecondSecond = false;
        hasSecondFirst = false;
    }

    /**
     * @return the secondYear
     */
    public SecondYear getSecondYear() {
        return secondYear;
    }

    /**
     * @param secondYear the secondYear to set
     */
    public void setSecondYear(SecondYear secondYear) {
        this.secondYear = secondYear;
    }

    /**
     * @return the firstYear
     */
    public FirstYear getFirstYear() {
        return firstYear;
    }

    /**
     * @param firstYear the firstYear to set
     */
    public void setFirstYear(FirstYear firstYear) {
        this.firstYear = firstYear;
    }

    public SecondYear getSecondSecondYear() {
        return secondSecondYear;
    }

    public void setSecondSecondYear(SecondYear secondSecondYear) {
        this.secondSecondYear = secondSecondYear;
    }

    public FirstYear getSecondFirstYear() {
        return secondFirstYear;
    }

    public void setSecondFirstYear(FirstYear secondFirstYear) {
        this.secondFirstYear = secondFirstYear;
    }

```

Match.java

```java
    public boolean isHasSecondSecond() {
        return hasSecondSecond;
    }

    public void setHasSecondSecond(boolean hasSecondSecond) {
        this.hasSecondSecond = hasSecondSecond;
    }

    public boolean isHasSecondFirst() {
        return hasSecondFirst;
    }

    public void setHasSecondFirst(boolean hasSecondFirst) {
        this.hasSecondFirst = hasSecondFirst;
    }

    @Override
    public String toString() {
        return "Match [secondYear=" + secondYear + ", secondSecondYear=" + secondSecondYear + ", firstYear=" + firstYear
                + ", secondFirstYear=" + secondFirstYear + ", hasSecondSecond=" + hasSecondSecond + ", hasSecondFirst="
                + hasSecondFirst + "]";
    }

}
```

Implementing Sort Algorithm:

Create and update Tally

```java
private void updateTally() {
    for (int i = 0; i < firstYears.size(); i++) {
        for (int j = 0; j < secondYears.size(); j++) {
            for (int k = 0; k < secondYears.get(j).getSelections().size(); k++) {
                if (matchFirsty(secondYears.get(j).getSelections().get(k))
                        && firstYears.get(i).equals(secondYears.get(j).getSelections().get(k))) {
                    tally.get(i).add(secondYears.get(j));
                    k = Integer.MAX_VALUE;
                }
            }
        }
    }
}
```

```java
private boolean matchFirsty(FirstYear firstYear) {
    for (int i = 0; i < firstYears.size(); i++) {
        if (firstYears.get(i).equals(firstYear)) {
            return true;
        }
    }
    return false;
}
```

The tally serves as a update of what every second years current top available choice is. This private helper method has been used to quickly and efficiently reevaluate the tally after matches have been made. This method may look complex but it is relatively simple. For every first year who has not been matched, this method searches for all the second years who have that firsty as

their current top pick. A current top pick is defined as the first year with the lowest position in a second years' selections array that has not already been picked.

Sort Algorithm

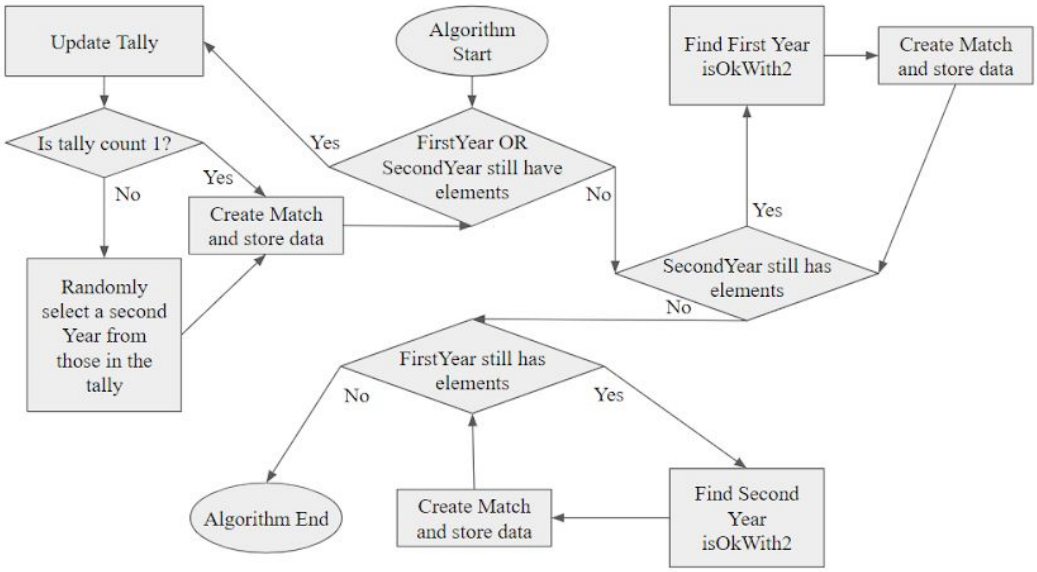
```
public void sort() {
    while (firstYears.size() > 0 && secondYears.size() > 0) {
        updateTally();

        for (int i = 0; i < tally.size(); i++) {
            if (tally.get(i).size() == 1) {
                matches.add(new Match(tally.get(i).get(0), firstYears.remove(i)));
                secondYears.remove(tally.get(i).get(0));
                updateTally();
                i--;
            } else if (tally.get(i).size() > 1) {
                int index = (int) Math.random() * tally.get(i).size();
                matches.add(new Match(tally.get(i).get(index), firstYears.remove(i)));
                secondYears.remove(tally.get(i).get(index));
                updateTally();
                i--;
            }
        }
    }
    while (firstYears.size() > 0) {
        for (int i = 0; i < firstYears.size(); i++) {
            for (int j = 0; j < matches.size(); j++) {
                if (matches.get(j).getSecondYear().isOkwith2()) {
                    matches.get(j).setSecondFirstYear(firstYears.remove(i));
                    matches.get(j).setHasSecondFirst(true);
                }
            }

        }
    }
    while (secondYears.size() > 0) {
        for (int i = 0; i < secondYears.size(); i++) {
            for (int j = 0; j < matches.size(); j++) {
                if (matches.get(j).getFirstYear().isOkwith2()) {
                    matches.get(j).setSecondSecondYear(secondYears.remove(i));
                    ;
                    matches.get(j).setHasSecondSecond(true);
                    matches.add(new Match(matches.get(j).getSecondSecondYear(), matches.get(j).getFirstYear(),
                            matches.get(j).getSecondYear()));

                }
            }
        }
```

This algorithm follows the flow chart shown in Criterion B Design: For reference that algorithm is shown again:


Update Tally
Algorithm Start
Find First Year isOkWith2
Create Match and store data
Is tally count 1?
Yes
FirstYear OR SecondYear still have elements
No
Yes
No
Create Match and store data
Yes
Randomly select a second Year from those in the tally
SecondYear still has elements
No
FirstYear still has elements
No
Yes
Algorithm End
Create Match and store data
Find Second Year isOkWith2