Week8Lab – 10 pts

Pre-lab questions

1. Why would we want to catch an exception?

   We may want to catch an exception to prevent a system crash for a minor issue such as the wrong type of data entered.

2. What might your program output when catching an IndexOutOfBoundsException?

   "The specified index value is outside the range of this list. Value should be between 0 and [List size - 1] inclusive."

3. Where does a 'throws' clause go? Write an example.

   This goes in the method header.

```
public static String findContactByEmail(ArrayList<Contact> contactList, String emailTarget)
throws NullPointerException
```

Choose one of the following to develop into a program that will allow you to practice with search and sort. Once chosen, do the following:

Understand the problem (restate in your own words, make any assumptions clear):

Pseudocode:

Name of file (.java) submitted:

Temperature Averages

Create an array of ints that represent temperatures. Find the highest and lowest temperature, print them out then calculate the range. Calculate and print the average temp for the timeframe represented by the array. Ask the user to select a range of temperatures by entering two ints, then give them the average in that span. Ex. User inputs 10 and 20, your program outputs the average temp for those 10 values. Since we can't control users, put this in a try/catch block in case they enter a number that is not an array index.

Gas prices

Create an array of doubles that represent gas prices. Find the highest and lowest price, print them out then calculate the range. Then sort the list in ascending order and find the middle value, print that out. Ask the user to select a gas price by entering a double and use binary search to find that value in the list. Tell the user where it is in the list and how much lower the lowest value is and how much higher the

highest value is. Since we can't control users, put this in a try/catch block in case they enter a number that is not in the array.

Dice rolls

Create an array of ints that represent the roll of a 6 sided die. Populate an array with 1000 simulated die rolls. Find the first roll for each number and print it out (ex. 1 @ index 3, 2 @ index 0, etc) using linear search. Sort the list and count how many of each roll there are (ex. 1 rolled 152 times, 2 rolled 166 times, etc). Shuffle the list then ask the user for an int between 0 and 999 and return the die roll at that location. Put this in a try/catch loop just in case.

We need to simulate 1000 dice rolls, find where the first roll of each value occurs (by list index), count how many rolls there are for each value by sorting the list first, shuffling the list and ask the user to print out a value at a given index in the data list making sure to handle any exceptions that may come up on data entry.

Files submitted: diceRolls.java

```
int [] rolls = new int [1000];


// Simulate the rolls
for i in 1…1000 do
        rolls[i-1] = random.uniformDist(1,6);


// Find where the first value occurs
int target = 1;
for i in 1…1000 do
        if rolls[i-1] == target then
                print("The first "+target+"occurs at index"+i)
                target += 1;


// Count the rolls by first sorting
sort(rolls);
int lowerBound = rolls[0];
```

```
int upperBound = sortedData[len(rolls)-1]

int [][] counts = new int [upperBound][2]

for roll in sortedData do
        counts[d-lowerBound][0] = d
        counts[d-lowerBound][1] += 1
for (cnt: counts)
        print(cnt[0]+": "+cnt[1]);

// Shuffle data and ask user to input the location of a roll
for i in 0…len(rolls-1) do
        int randIndex =  random.uniform(0, data.length);
        int temp = data[randIndex];
        data[randIndex] = data[i];
        data[i] = temp;
done = false
while not done do
        try:
                int index = input("Enter a value between 0 - 999: ")
                printf("The roll at index %d is a %d.\n", index, rolls[index])
                done = true
        catch IndexOutOfBoundsException i:
                print("Index not in range");
        catch InputMismatchException p:
                print("Invalid entry")
```