Collin Gros
10-06-2020
cs471

# JAVA EXCEPTION HANDLING ASSIGNMENT

## PROBLEM DESCRIPTION

Re-write 'grade_distribution.adb' in Java, but only assign values to freq[] in a catch block (whenever an exception occurs).

## CODE

```
/*
	collin gros
	10-06-2020
	cs471
	exception handling in java

	this code is a rewritten version of 'grade_distribution.adb', which
	was a code snippet written in ADA to count frequencies of grade
	ranges of 10, between 0 and 100. the ADA version had some exception
	handling, but this was extended in this version to only modify the
	freq[] array in a catch section of a try{}catch{} block.
*/
import java.util.Scanner;
import java.util.InputMismatchException;
import java.lang.ArithmeticException;


public class GradeDist
{
	public static void main(String[] args)
	{
		// scanner for reading user input
		Scanner scan = new Scanner(System.in);
		// 0-9 instead of 1-10 and all are already
		// initialized to 0 (because of java arrays)
		int[] freq = new int[10];

		// grab user input for freqs. will go to catch block if
		// a non-natural number is entered for n, or the number is
		// out of the range [0, 100]
		//
		// otherwise, it just incrememements the value under that
		// frequency index
		int n = 0;
		while (true) {
			try {
				n = scan.nextInt();
				// if n is negative, it's not a natural number
				if (n < 0) {
					throw new InputMismatchException();
				}

				// set off div by zero exception so that we can
				// modify freq, as per the instructions
```

```
                        throw new ArithmeticException();
                    }
                    // everything is fine; this is thrown on purpose to assign
                    // to freq
                    catch (ArithmeticException outside_e) {
                        // calculate index
                        int idx = n / 10;
                        try {
                            freq[idx] = freq[idx] + 1;
                        }
                        // idx was out of range
                        catch (ArrayIndexOutOfBoundsException inside_e) {
                            if (n == 100) {
                                freq[9] = freq[9] + 1;
                            }
                            else {
                                System.out.println("Error -- new grade: " + n
+
                                                   " is " + "out of
range");
                            }
                        }
                    }
                    // n was not an integer (breaks the loop and prints results)
                    catch (InputMismatchException e) {
                        System.out.println("Limits\tFrequency\n");
                        // loop iterates from freq[0] to freq[8] and prints value
                        // in the proper distribution range
                        //
                        // freq[9] is skipped because it was skipped in the ada
                        // code. i don't know if this was a mistake or what but
                        // i'm going to emulate as close to the ada code as
                        // i can
                        for (int i = 0; i < 9; ++i) {
                            int limit1 = 10 * i;
                            int limit2 = limit1 + 9;
                            if (i == 9) {
                                limit2 = 100;
                            }
                            System.out.println(limit1 + "\t" + limit2 + "\t"
                                               + freq[i]);
                        }

                        // WE'RE OUTTA HERE!
                        break;
                    }
                }
            }
}
```

## TEST

| ADA | JAVA |
| --- | --- |

```
$ ./grade_distribution
10
25
30
100
0
101
Error -- new grade:         101
is out of range
-1
Limits    Frequency

          0          9          1
         10         19          1
         20         29          1
         30         39          1
         40         49          0
         50         59          0
         60         69          0
         70         79          0
         80         89          0
```

```
$ java GradeDist
10
25
30
100
0
101
Error -- new grade: 101 is out of
range
-1
Limits     Frequency

0    9    1
10   19   1
20   29   1
30   39   1
40   49   0
50   59   0
60   69   0
70   79   0
80   89   0
```