HW2Code

Divisors.java:

```
public class Divisors {
	public static void main (String[] args) {
		int number = Integer.parseInt(args[0]);
		for (int i = 1; i < (number / 2) + 1; i++ ) {
			if (number % i == 0) {
				System.out.println(i);
			}
		}
		System.out.println(number);
	}
}
```

Reverse.java:

```
public class Reverse {
	public static void main (String[] args){
		String word = args[0];
		for (int i = 0; i < word.length(); i++) {
			if ((i + 1) == word.length()) {
				System.out.println(word.charAt(word.length() - 1 - i));
			}else{
				System.out.print(word.charAt(word.length() - 1 - i));
			}
		}
		if (word.length() % 2 == 0) {
			System.out.println("The middle character is " +
				word.charAt((word.length() / 2) - 1));
		}else{
			System.out.println("The middle character is " +
				word.charAt(word.length() / 2));
		}
	}
}
```

InOrder.java:

```
public class InOrder {
	public static void main (String[] args) {
		int number = (int)(Math.random() * 10);
		int previousNumber = 0;
		while (previousNumber <= number) {
			previousNumber = number;
			System.out.print(number + " ");
			number = (int)(Math.random() * 10);
		}
	}
}
```

DamkaBoard.java:

```
public class DamkaBoard {
	public static void main(String[] args) {
		int n = Integer.parseInt(args[0]);
		int j = 0;
		int i = 0;
		while ( i < n ) {
			while ( j < n ) {
				j++;
				if (i%2 != 0) {
					if (j == n) {
						System.out.println(" *");
					}else{
						System.out.print(" *");
					}
				}else{
					if (j == n) {
						System.out.println("* ");
					}else{
						System.out.print("* ");
					}
				}
			}
			i++;
			j = 0;
		}
	}
}
```

Perfect.java:

```
public class Perfect {
	public static void main (String[] args) {
		int number = Integer.parseInt(args[0]);
		int perfectNumber = 0;
		int[] perfectNumberArray = new int[1000000];
		int amountOfPerfect = 0;
		for (int i = 1; i < (number / 2) + 1; i++) {
			if (number % i == 0) {
				perfectNumber = perfectNumber + i;
				perfectNumberArray[amountOfPerfect] = i;
				amountOfPerfect++;
			}
		}
		if (perfectNumber == number) {
			System.out.print(number + " is a perfect number since " + number +
				" = " );
			for (int j = 0; j < amountOfPerfect; j++) {
				if( j == 0){
					System.out.print(perfectNumberArray[j]);
				}else{
					System.out.print(" + " + perfectNumberArray[j]);
				}
			}
		}else{
			System.out.println(number + " is not a perfect number");
		}
	}
}
```

OneOfEachStats.java:

```
import java.util.Random;

public class OneOfEachStats {
	public static void main (String[] args) {
		int T = Integer.parseInt(args[0]);
		int seed = Integer.parseInt(args[1]);
		Random generator = new Random(seed);
		double totalAmountOfChildren = 0;
		int amountOfFamilies = 0;
		int mostCommonAmountOfChildren = 0;
		int[] familySize = new int[5];
		for (int i = 0; i < T; i++ ) {
			boolean noGirl = true;
			boolean noBoy = true;
			int amountOfChildren = 0;
			while (noBoy || noGirl) {
				double randomNumber = generator.nextDouble();
				amountOfChildren++;
				if (randomNumber >= 0.5) {
					noGirl = false;
				}else{
					noBoy = false;
				}
			}
			totalAmountOfChildren = totalAmountOfChildren +
				amountOfChildren;
			if (amountOfChildren > 4) {
				familySize[4] = familySize[4] + 1;
			}else{
				familySize[amountOfChildren] =
					familySize[amountOfChildren] + 1;
			}
		}

		for (int j = 0; j < familySize.length; j++ ) {
			if (familySize[j] > amountOfFamilies) {
				amountOfFamilies = familySize[j];
				mostCommonAmountOfChildren = j;
```

```
                }
            }

            System.out.println("Average: " + ( totalAmountOfChildren / T ) + " children
                to get at least one of each gender.");
            System.out.println("Number of families with 2 children: " + familySize[2]);
            System.out.println("Number of families with 3 children: " + familySize[3]);
            System.out.println("Number of families with 4 or more children: " +
                familySize[4]);
            if (mostCommonAmountOfChildren > 4) {
                System.out.println("The most common number of children is 4 or
                    more.");
            }else{
                System.out.println("The most common number of children is " +
                    mostCommonAmountOfChildren + ".");
            }
        }
}
```