HW2Code

DamkaBoard:

```
public class DamkaBoard {
   public static void main(String[] args) {
      int n = Integer.parseInt(args[0]);
      for(int i = 1; i <= n; i++){
         for(int j = 1; j<=n; j++){
            if(i%2 == 0){
            System.out.print(" *");
            } else {
            System.out.print("* ");
            }
         }
         System.out.println();
      }

   }

}
```

Divisors:

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


         }


       }
     }

  }
```

InOrder:

```
public class InOrder {
   public static void main (String[] args) {
       int x = (int)(Math.random()*10);
       int xPrev = x;

      do{
         System.out.print(" " + x);
         xPrev = x;
         x = (int)(Math.random()*10);
       } while( x >= xPrev);

   }
}
```

OneOfEach:

```
public class OneOfEach {
   public static void main (String[] args) {
      int g = 0;
      int b = 0;
      do{
         double x = Math.random();
         if (0<= x && x < 0.5) {
            System.out.print("g "+ " ");
            g++;
          }else {
            System.out.print("b " + " ");
            b++;
         }
      }while ( g==0 || b==0 );

      System.out.println("You made it... and you now have " + (g+b) + " children");
      }

   }
```

OneOfEachStats:

```
public class OneOfEachStats {
   public static void main (String[] args) {
      // Gets the two command-line arguments
      int n = Integer.parseInt(args[0]);
      int seed = Integer.parseInt(args[1]);
      // Initailizes a random numbers generator with the given seed value
      Random generator = new Random(seed);
      int sumbg = 0;
      int twoChildren = 0;
      int threeChildren = 0;
      int fourChildren = 0;
      for( int i = 0; i<n; i++){
         int g = 0;
         int b = 0;

         do{
            double x = generator.nextDouble();
            if (0<= x && x < 0.5) {
            //System.out.print("g "+ " ");
               g++;
            }else {
            //System.out.print("b " + " ");
               b++;
            }
         }while ( g==0 || b==0 );
         sumbg = sumbg + (g+b);
         if(g+b == 2){
```

```
            twoChildren++;
          }
          if(g+b == 3){
            threeChildren++;
          }
           if(g+b >= 4){
            fourChildren++;
           }
        }
        System.out.println("Average: " + ((double)sumbg)/n +  " children to get at least one of each gender.");
        System.out.println("Number of families with 2 children: " + twoChildren);
        System.out.println("Number of families with 3 children: " + threeChildren);
        System.out.println("Number of families with 4 or more children: " + fourChildren);
        int commonNumber = Math.max(twoChildren,Math.max(threeChildren,fourChildren));
        if(commonNumber == twoChildren){
          System.out.println("The most common number of children is 2.");
        }
        if(commonNumber == threeChildren){
          System.out.println("The most common number of children is 3.");
        }
        if (commonNumber == fourChildren) {
          System.out.println("The most common number of children is 4.");
        }
```