HW02 – Romi Golan

**Divisors**

```
public class Perfect
{
	public static void main (String[] args)
	{
		int num = Integer.parseInt(args[0]);
		int sum = 1;

		String pelet = "";

			for(int i = 2; i < num; i++)
			{
				if(num % i == 0)
				{
					pelet = pelet + " + " + i;
					sum = sum + i;
				}
			}

			if(sum == num)
				{
					System.out.print(num + " is a perfect number since " +
					num + " = 1");
					System.out.print(pelet);
				}
			else
				{
					System.out.print(num + " is not a perfect number");
				}
	}
}
```

**Reverse**

```
public class Reverse
{
	public static void main (String[] args)
	{
		String word = args[0];

		int middle;
		char outLetter;

		if(word.length() % 2 == 0)
			{
				middle = (word.length() + 1) / 2;
				outLetter = word.charAt(middle - 1);
			}
		else
			{
				middle = word.length() / 2;
				outLetter = word.charAt(middle);
			}

		int i = word.length()-1;

		while(i >= 0)
		{
			char letter = word.charAt(i);
			System.out.print(letter);
			i--;
		}

		System.out.println();
		System.out.println("The middle character is " + outLetter);
	}
}
```

**InOrder**

```
public class InOrder
{
	public static void main (String[] args)
	{
		int biggestNum = ((int)(Math.random() * 10));
		System.out.print(biggestNum);

		int num;

		while(biggestNum < 9)
			{
				num = ((int)(Math.random() * 10));
				if(num >= biggestNum)
					{
						System.out.print(" " + num);
						biggestNum = num;
					}
			}

	}
}
```

**DamkaBoard**

```
public class DamkaBoard
{
      public static void main(String[] args)
      {
            int num = Integer.parseInt(args[0]);

            for(int i = 1; i <= num; i++)
            {


                  for(int j = 1; j <= num; j++)
                  {
                        if(i % 2 == 0)
                        {
                        System.out.print(" *");
                        }
                        else
                        {
                        System.out.print("* ");
                        }
                  }
                  System.out.println();
            }
      }
}
```

**Perfect**

```
public class Perfect
{
      public static void main (String[] args)
      {
            int num = Integer.parseInt(args[0]);
            int sum = 1;

            String pelet = "";

                  for(int i = 2; i < num; i++)
                  {
                        if(num % i == 0)
                        {
                              pelet = pelet + " + " + i;
                              sum = sum + i;
                        }
                  }

                  if(sum == num)
                        {
                              System.out.print(num + " is a perfect number since " +
                              num + " = 1");
                              System.out.print(pelet);
                        }
                  else
                        {
                              System.out.print(num + " is not a perfect number");
                        }
      }
}
```

**OneOfEachStats**

```
import java.util.Random;
public class OneOfEachStats
{
	public static void main (String[] args)
	{
		int families = Integer.parseInt(args[0]);
		int seed = Integer.parseInt(args[1]);
		Random generator = new Random(seed);
		int i = 1;

		int boys = 0;
		int girls = 0;

		boolean enough = false;

		int sumKidsAvrege = 0;

		int TwoMembers = 0;
		int ThreeMembers = 0;
		int FourOrMoreMembers = 0;

		while(i <= families)
		{
			while(enough == false)
			{
				double boyORgirl = (int)(generator.nextDouble()*2);

				if(boyORgirl == 0)
				{
					girls++;
				}
				else
				{
					boys++;
				}

				if((boys > 0) && (girls > 0))
				{
					enough = true;
					sumKidsAvrege = sumKidsAvrege + boys + girls;
					if((boys + girls) < 3)
						{
							TwoMembers++;
						}
						else
						{
							if((boys + girls) == 3)
```

```
                                    {
                                          ThreeMembers++;
                                    }
                                    else
                                    {
                                          FourOrMoreMembers++;
                                    }
                              }
                        }
                  }
                  i++;
                  enough = false;
                  boys = 0;
                  girls = 0;
            }
            double FamForAvg = families;

            System.out.println("Average: " + (sumKidsAvrege/FamForAvg) + "
            children to get at least one of each gender.");

            System.out.println("Number of families with 2 children: " +
            TwoMembers);
            System.out.println("Number of families with 3 children: " +
            ThreeMembers);
            System.out.println("Number of families with 4 or more children: " +
            FourOrMoreMembers);

            if((TwoMembers >= ThreeMembers) && (TwoMembers >= FourOrMoreMembers))
                  {
                        System.out.println("The most common number of children is
                        2.");
                  }
                  else
                  {
                        if(ThreeMembers >= FourOrMoreMembers)
                        {
                              System.out.println("The most common number of children
                              is 3.");
                        }
                        else
                              System.out.println("The most common number of children
                              is 4 or more.");
                  }
      }
}
```