Labs 3 and 4 (Week 2)
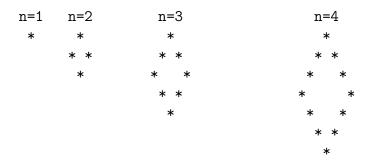
1. Read two dates in the format `dd mm yyyy` and print the one which is earlier.

2. Let us call a number (positive integer) *simple* if all its prime divisors are from the set $\{2, 3, 5\}$. 1 is considered *simple* by definition. The first 11 *simple* numbers are $1, 2, 3, 4, 5, 6, 8, 9, 10, 12, 15$. You have to write a program that reads a positive integer n and prints out the nth *simple* number.

3. In school we learnt simple tests to check for divisibility of a number by $2, 3, 5$ etc. Write a divisibilitiy tester that takes two positive integers m and n and checks if m is divisible by n using the following rules. We assume n must belong to the set $\{2, 3, 5, 6, 8, 11\}$. The divisibility rules are:

   - m is divisible by 2 if its last digit is divisible by 2.
   - m is divisible by 3 if the sum of its digits is divisible by 3.
   - m is divisible by 5 if its last digit is either 5 or 0.
   - m is divisible by 6 if it is divisible by 2 and by 3.
   - m is divisible by 8 if the number formed by its last 3 digits is divisible by 8.
   - m is divisible by 11 if the sum of all digits in the odd position is equal to the sum of all digits in the even position. For example: 121, 1232.

   Your program should print YES if m is divisible by n and n is a legal divisor - that is belongs to the set $\{2, 3, 5, 6, 8, 11\}$. If n is legal and it does not divide m print NO and if n is not a legal divisor print -1.

4. Write a program to read a positive integer n and print an nth order pattern. The patterns for $n = 1, 2, 3, 4$ are shown below.

```
n=1    n=2          n=3                   n=4
 *      *            *                     *
       * *          * *                   * *
        *          *   *                 *   *
                    * *                 *     *
                     *                  *     *
                                         * *
                                          *
```

5. Given two dates in the format *dd mm yyyy* find the number of days between the two dates. If the two dates are same then we count the number of days in between as 1. This problem is harder and requires more code. You have to take care of leap years as well.