# 3 Triangle

**Competency 3 (CLO 1, CLO 2)** Students can construct recursive predicates in Prolog. Students can perform arithmetic operations in Prolog.

**Remark 3** This problem is worth **15 points**. The problem is adapted from the 1995 Prolog Programming Contest, in Portland, USA.

Write a predicate `triangle/1` which requires an argument $N$ and it succeeds whenever $N$ is a non-negative integer, `triangle(N)` also yields a *triangle-shaped* string of $N$ rows. For each row $i$, there are $i$ characters of star (`*`), every two stars are separated by a space, and every first star of that row is preceded by $N - i - 1$ blank spaces. For example:

- `?- triangle(1).` returns

```
*
```

- `?- triangle(4).` returns

```
      *
     * *
    * * *
   * * * *
```

- `?- triangle(5).` returns

```
        *
       * *
      * * *
     * * * *
    * * * * *
```

**Hint 3** Modify the idea of solving Problem 2 (Triangles) in Homework 3. In addition, suppose the significant leading space is indicated by an underscore _, we have

```
_ _ _ _ *          _ _ _ _          *
_ _ _ * *          _ _ _            * *
_ _ * * *     =    _ _         +    * * *
_ * * * *          _                * * * *
* * * * *                           * * * * *
```

(Note: some side effects, such as **true** or **false** are admissible. However, your program should avoid any infinite recursive call.)