# SEQUENCE PUZZLE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***N*** | ***Base*** | | ***Base + Implied*** | |
|  | *Fails* | *Time* | *Fails* | *Time* |
| **500** | 617 | 26s 136msec | 495 | 16s 246msec |
| **1000** | 1247 | 3m 1s | 995 | 1m 23s |

*What is happening when going base → base+implied ? Why?*

The table shows the failures and total processing time of the two models implemented for the “A Sequence Puzzle” problem.

The difference between the two models is in the use of two implicit constraints: this causes a variation on failures and on the time used for code execution. Implicit constraints do not cause a change in the solution set, but they can significantly reduce the search space. The constraints inserted only provide additional information and create greater propagation for the model which allows it to arrive at the solution more quickly.

In the first model the failures are higher than in the second, this is caused by the constraints present in the code which reduce the search space.

As you can see, the second model reduces the processing time compared to the first, although for n=1000 the time is still high, presumably because there are more decision variables and a wider range of values from which to select for each variable.