BDSA Assignment 00

Tobias G. Spoorendonk

September 8, 2022

1 Introduction

This is a short report on the solution to the mandatory assignment in the course, BSC: Analysis, Design and Software Architecture, as prompted by exercise 7 of the assignment. This report will be discussing the implementation of an algorithm to compute whether a specific year was a leap year according to the definition of the Gregorian Calendar.

2 Algorithmic Requirements

The requirements given by the exercise states that:

1. Every fourth year is a leap year
2. Point 1 does NOT apply if the year is also divisble by 100
3. Point 2 does not apply if the year is also divisible by 400
4. Finally the Gregorian Calendar only takes into account years from 1582 AD and after, thus any years before this do not make sense in the context of the algorithm

This means that year 1200 is not a valid input, year 1604 is a leap year, year 1700 is not a leap year, while year 2000 is one.

3 implementation

The algorithm used by the program roughly corresponds to Figure 1. There are some differences in the source code for the project compared to what is actually shown on the diagram which are abstracted into a more intuitive look in the flow chart. These discrepancies are mostly due to optimisations and ease of coding (For example, the actual algorithm is constantly keeping track of the current state of the year, leap year or not, and flips from one state to the other until it finishes all checks, where it finally decides if the year is a leap year or not. This generalised in the flow chart despite not being completely identical).

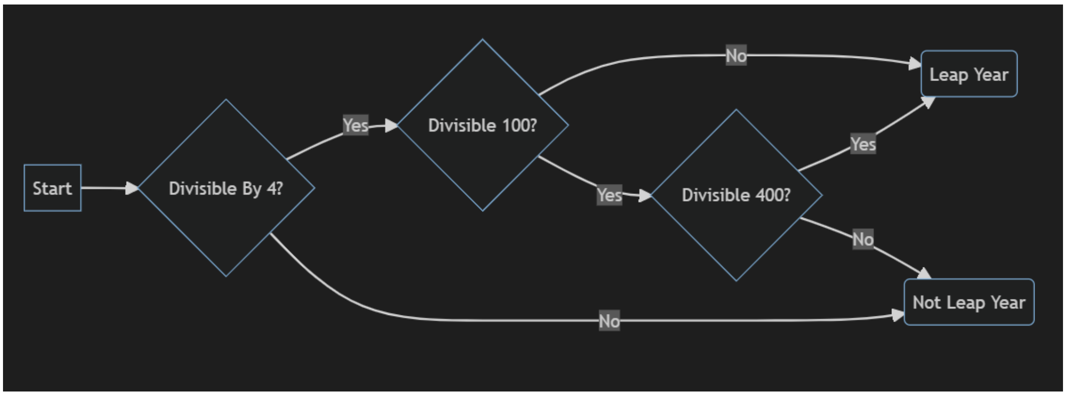

Figure 1: A Flow diagram representing the implementation of the algorithm to determine whether a year is a leap year or not.