

# ETERNITY: NUMBERS - PI

Baiyu Huo <sup>†</sup>

<sup>†</sup> SOEN 6481 - Software System Requirements Specification

## Introduction

The project is about making a calculator which computes the eternity number PI from scratch and provides some related applications which related to PI. The whole project contains the following phrases:

- Understanding the problem domain.
- Collecting the requirements by performing an interview with expertise.
- Documented the requirements (use case, UML diagram, maintain the requirement traceability ).
- Implementation.

The primary task in this project is about the PI and its functions. PI is a mathematical constant which people has been doing research on for a long time and being applied in a lot of famous mathematical problem solutions. In this project, the PI was calculated in 2 different algorithms and 6 different degrees of precision. Besides the general functions in the calculator, The calculator provides "calculate the area of a circle" and "calculate the circumference of a circle" — two extra functions that need to use the PI.

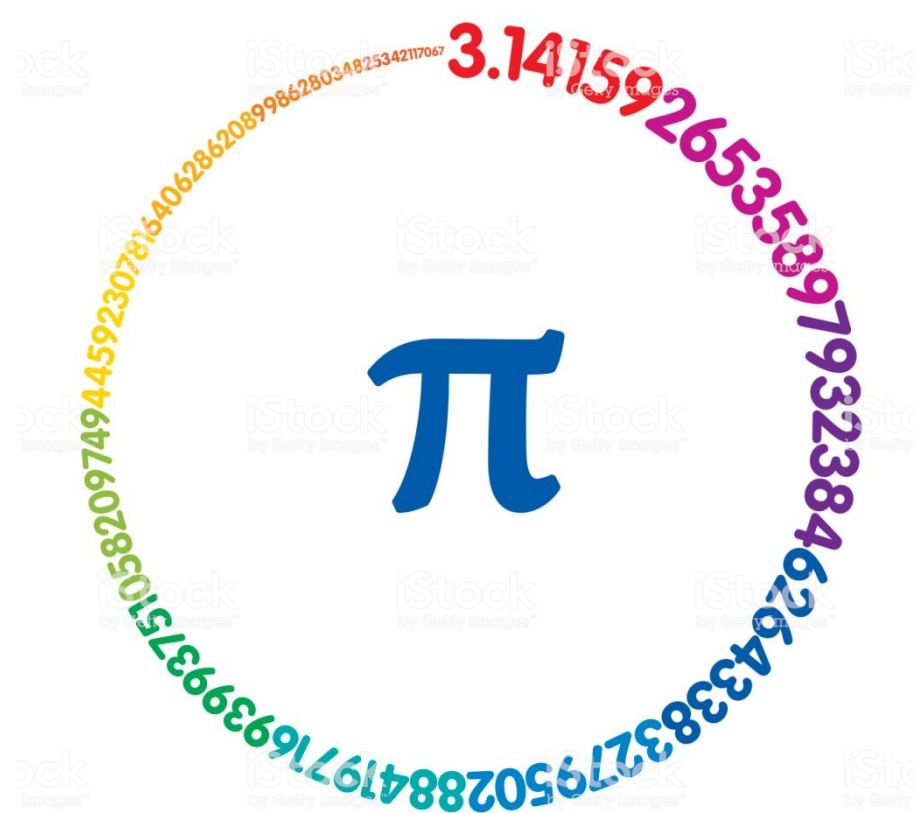


Fig. 1: PI (Source: Google)

## Critical Decisions

### 1 The Selection of Interviewee

Because PI appears in many formulas in all areas of mathematics and physics, the choices of the interviewee in this project have to narrow down to the people who have mathematics and physics background or work in related fields. Dr. Wang and Ms. Xuan are two candidates for the interview. Dr. Wang is a mathematical course lecturer in college, Ms. Xuan, on the other hand, works as a bank accounting with a bachelor in mathematics degree. At this stage, the character of the candidates became a very important factor. Finally, Dr. Wang was chosen due to the his spontaneous nature which eased the flow of the interview.

### 2 Decisions about the functions

Besides the general functions in the calculator, there are a lot of applications which use PI in the user story such as "circle area calculation", "circle circumference calculation" and "Trigonometry calculation". According to the time limitation, only one or two of them can be implemented. The final decision was made in terms of the degree of relation and, implementing efforts and the feedback from the interviewee which also be considered as the customer of the project. For implementing the trigonometry calculation, the function of sin, cosine and tan need to be implemented, which was hard to finish within the project period, so the area calculation and circumference had been chosen.

## Critical Decisions

### 3 The Choice Between Precision or Performance

Computing the PI is one of most important functions in this project. However, the more accurate pi being calculated, the more time it needs, which leads to low performance. But, both accuracy and performance are important in a different scenario. After analyzing each of the scenarios, the final decision is to give an extra function to users to choose the degree of precision of pi, so that they can tackle the problem in different situations.



Fig. 2: Decision among qualities (Source: Google)

### 4 Trade offs for architecture

The project was designed to have multiple qualities such as high extensibility, high maintainability, which requires the component to achieve modularity, high cohesion and low coupling. To achieve those features, a set of design patterns are applied in this project, such as singleton pattern, memento pattern and strategy pattern. It increased the extensibility of the program and increased the understandability for the maintainer. However, applied too many unnecessary patterns is also not a good thing, so there are some trade-offs about the architecture of the project. The pattern would only be applied when certain qualified need to be achieved. What's more, the project is designed under the SOLID principles, especially the open-close principles.

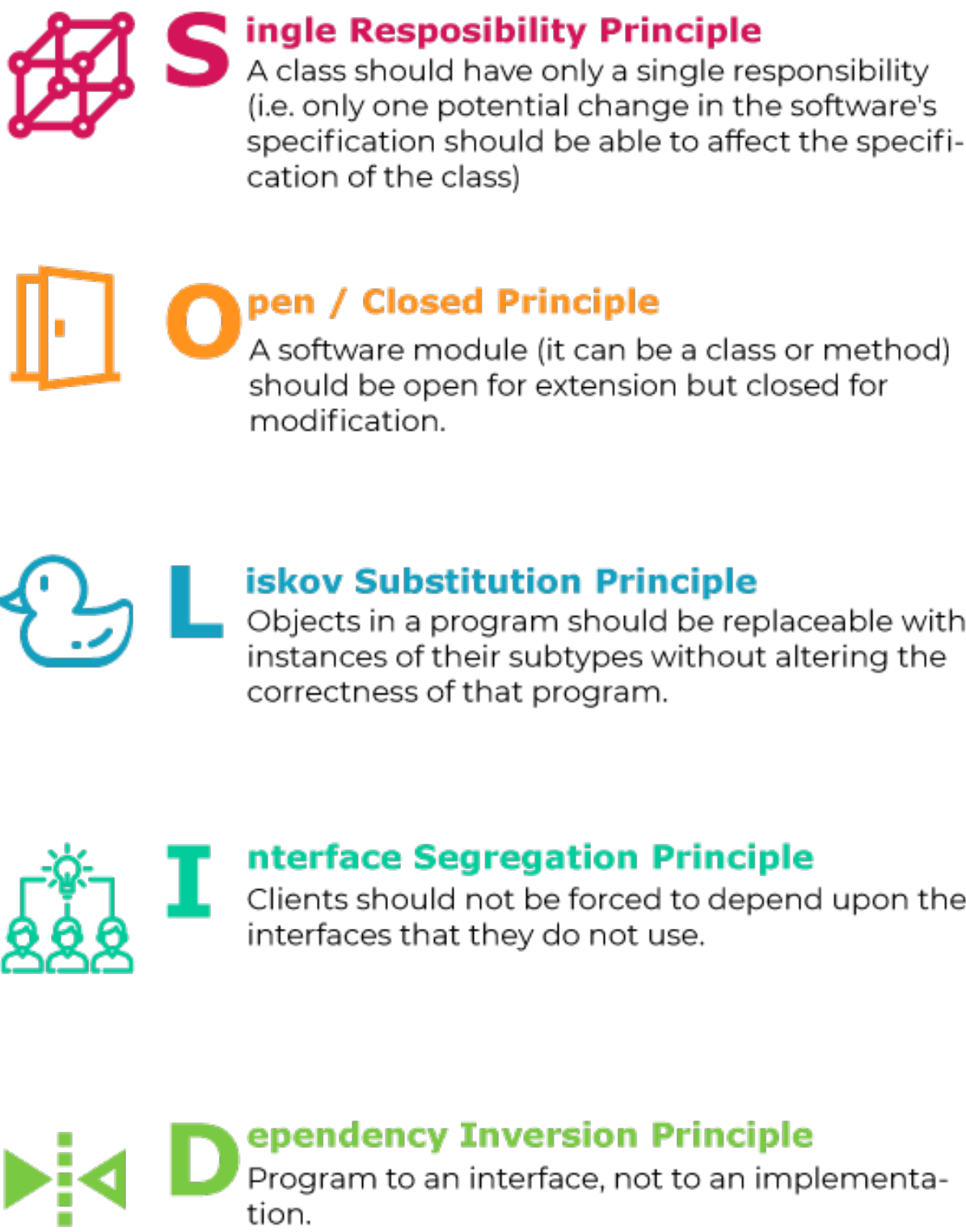


Fig. 3: SOLID (Source: Devopedia)

## Challenges

### 5 Time Limitation

As the length of the project is only 3 weeks, it is a kind of challenge to understand the knowledge about requirement engineering and applied them in the project. Since the software is a process, as we mentioned at the beginning of the poster, there are 4 phases in the project which map the different phase of the software life circle except for the maintenance phases.

### 6 Understand Mathematical problem domain

Because I don't have any background in mathematics, so the calculation of pi is one of the main challenges for me. After research, I got two algorithms from the internet, however, it is hard for me to understand the theory of it.

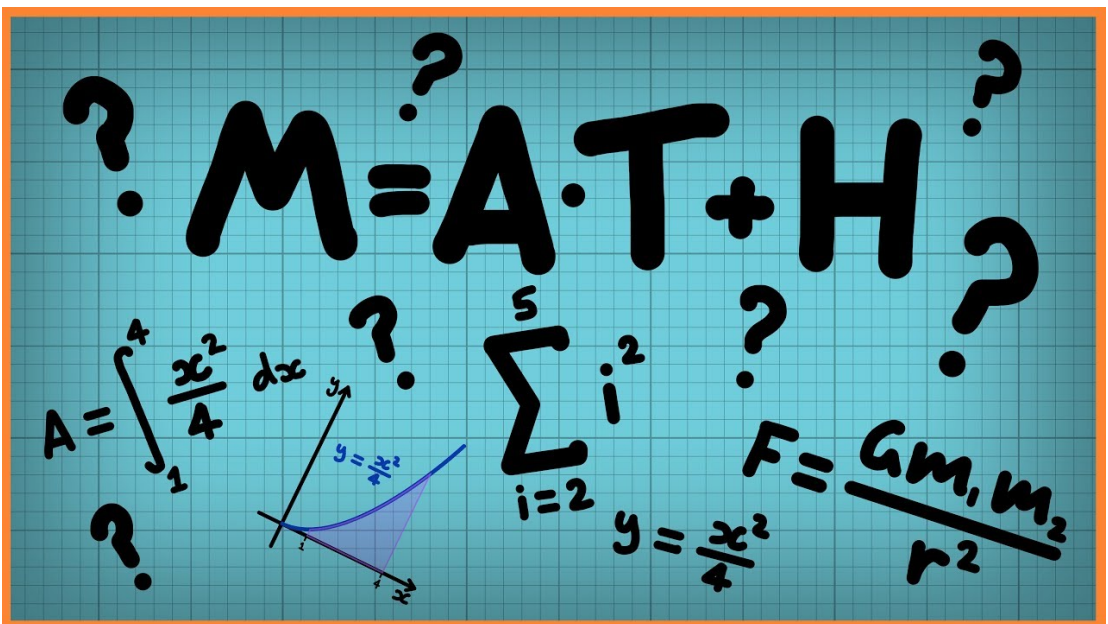


Fig. 4: Math (Source: Youtube)

### 7 Implementation of math function in java

According to the instruction of the project, we can not able to used any build-in functions in java.math package, so we implement the functions such as Square root, power and PI.

## Lessons

- Embrace the challenges, even that means you need to jump out of your comfort zone.
- Never put off until tomorrow what you can do today.
- Communication is really important in software requirement.
- You need to understand the problem domain first, before making the decisions during the software engineering process.

## References

- 1 Wikipedia contributors. Pi — Wikipedia, the free encyclopedia, 2019. [Online; ac-cessed 6-August-2019].
- 2 Wikipedia contributors. SOLID — Wikipedia, the free encyclopedia, 2019. [Online; ac-cessed 7-August-2019]
- 3 Projectsmart contributors. understanding-the-project-management-triple-constrain, 2019. [Online; ac-cessed 7-August-2019]