

# CPSC-354 Report

Jeffrey Bok  
Chapman University

September 1, 2025

**Abstract**

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Week by Week</b>	<b>2</b>
2.1	Week 1 . . . . .	2
2.1.1	Homework . . . . .	2
2.1.2	Exploration . . . . .	2
2.1.3	Questions . . . . .	3
2.2	Week 2 . . . . .	5
2.2.1	Homework . . . . .	5
2.2.2	Exploration . . . . .	5
2.2.3	Questions . . . . .	5
2.3	Week 3 . . . . .	5
2.3.1	Homework . . . . .	5
2.3.2	Exploration . . . . .	5
2.3.3	Questions . . . . .	5
2.4	Week 4 . . . . .	5
2.4.1	Homework . . . . .	5
2.4.2	Exploration . . . . .	5
2.4.3	Questions . . . . .	5
2.5	Week 5 . . . . .	5
2.5.1	Homework . . . . .	5
2.5.2	Exploration . . . . .	5
2.5.3	Questions . . . . .	5
2.6	Week 6 . . . . .	5
2.6.1	Homework . . . . .	5
2.6.2	Exploration . . . . .	5
2.6.3	Questions . . . . .	5
2.7	Week 7 . . . . .	5
2.7.1	Homework . . . . .	5
2.7.2	Exploration . . . . .	5
2.7.3	Questions . . . . .	5
2.8	Week 8 . . . . .	5
2.8.1	Homework . . . . .	5
2.8.2	Exploration . . . . .	5

2.8.3	Questions	5
2.9	Week 9	5
2.9.1	Homework	5
2.9.2	Exploration	5
2.9.3	Questions	5
2.10	Week 10	5
2.10.1	Homework	5
2.10.2	Exploration	5
2.10.3	Questions	5
2.11	Week 11	5
2.11.1	Homework	5
2.11.2	Exploration	5
2.11.3	Questions	5
<b>3</b>	<b>Synthesis</b>	<b>5</b>
<b>4</b>	<b>Evidence of Participation</b>	<b>5</b>
<b>5</b>	<b>Conclusion</b>	<b>5</b>

# 1 Introduction

## 2 Week by Week

### 2.1 Week 1

#### 2.1.1 Homework

What is the MU Puzzle and how do you "solve" it?:

The MU puzzle is a logic puzzle created by Douglas Hofstadter in his 1979 book "Gödel, Escher, Bach: An Eternal Golden Braid." It's designed to illustrate concepts about formal systems, computability, and the limits of rule-based reasoning. The rules are below:

Rule I: If a string ends in I, you can add U to the end ( $xI \rightarrow xIU$ )

Rule II: If you have Mx, you can make Mxx (double everything after M)

Rule III: If you find III anywhere in your string, you can replace it with U ( $xIIIy \rightarrow xUy$ )

Rule IV: If you find UU anywhere in your string, you can remove it ( $xUUy \rightarrow xy$ )

To "solve" the puzzle, you try to apply a combination of rules step by step, creating new strings. Eventually, you'll find that MU can never be reached because the rules never allow you to remove the odd number of I's needed to get zero.

#### 2.1.2 Exploration

Hofstadter used this puzzle to demonstrate how formal systems can have inherent limitations - some statements that seem like they should be provable within a system are actually unprovable. This connects to Gödel's incompleteness theorems and fundamental questions about the nature of mathematical truth and computation.

Programming languages are formal systems, just like the MU puzzle. They have:

- Syntax rules (what constitutes valid code)
- Transformation rules (how expressions evaluate)

- Semantic constraints (what programs can actually compute)

The MU puzzle demonstrates that even simple rule sets can have hidden limitations - similarly, programming languages have inherent computational boundaries.

### **2.1.3 Questions**

#### **1. TEST QUESTIONS**



- 2.2 Week 2
  - 2.2.1 Homework
  - 2.2.2 Exploration
  - 2.2.3 Questions
- 2.3 Week 3
  - 2.3.1 Homework
  - 2.3.2 Exploration
  - 2.3.3 Questions
- 2.4 Week 4
  - 2.4.1 Homework
  - 2.4.2 Exploration
  - 2.4.3 Questions
- 2.5 Week 5
  - 2.5.1 Homework
  - 2.5.2 Exploration
  - 2.5.3 Questions
- 2.6 Week 6
  - 2.6.1 Homework
  - 2.6.2 Exploration
  - 2.6.3 Questions
- 2.7 Week 7
  - 2.7.1 Homework
  - 2.7.2 Exploration
  - 2.7.3 Questions
- 2.8 Week 8
  - 2.8.1 Homework
  - 2.8.2 Exploration
  - 2.8.3 Questions
- 2.9 Week 9
  - 2.9.1 Homework
  - 2.9.2 Exploration
  - 2.9.3 Questions
- 2.10 Week 10
  - 2.10.1 Homework
  - 2.10.2 Exploration
  - 2.10.3 Questions
- 2.11 Week 11
  - 2.11.1 Homework