Creative Coding

Lab #6

Name, Student Number

*TU850/1: BSc (Hons.) Data Science*

*and Artificial Intelligence*

Table of Contents

[1. Introduction & Methodology 3](#_Toc194702890)

[1.1. Introduction 3](#_Toc194702891)

[1.2. Tools and Setup 3](#_Toc194702892)

[1.3. Methodology 3](#_Toc194702893)

[2. Experiments and Outcomes 4](#_Toc194702894)

[2.1. Figure 1: Four Squares 4](#_Toc194702895)

[2.2. Figure 2: Interlocked Circles 4](#_Toc194702896)

[2.3. Figure 3: Alien Character 5](#_Toc194702897)

[2.4. Figure 4: Shamrock 5](#_Toc194702898)

[3. Reflection, Challenges & Learnings 6](#_Toc194702899)

[3.1. Challenges 6](#_Toc194702900)

[3.2. Conclusion 6](#_Toc194702901)

[Appendix A: My Code 7](#_Toc194702902)

[A1. Squares Code 7](#_Toc194702903)

[A2. Olympic Rings code 8](#_Toc194702904)

[A3. Space Invader alien code 9](#_Toc194702905)

[A4. Shamrock code 10](#_Toc194702906)

## Introduction & Methodology

### Introduction

This report reflects on my experiments, learning process, and outcomes for Creative Coding Assignment #1. The assignment required me to recreate four different diagrams using Python’s Turtle graphics module, a tool that provides a visual and interactive way to understand programming concepts. Each figure presented a unique set of challenges, testing various aspects of drawing logic and control structures. The process involved geometric reasoning to break down complex patterns into simpler components that could be reproduced algorithmically. Additionally, I considered visual aesthetics, symmetry, and composition, learning how to balance creativity with precision in code-based art. Overall, this assignment deepened my understanding of both the technical and artistic sides of generative programming.

### 

### Tools and Setup

* Programming Language: Python 3.10
* Library Used: turtle
* Environment: PyCharm
* GenAI Tool: XXXXXXXXX

### Methodology

For each diagram, I followed this process:

* **Observation**: Closely studied the example figure provided in the assignment.
* **Decomposition**: Broke the image into shapes, calculating coordinates, angles, and relative positions.
* **Experimentation**: Used trial-and-error in Turtle to draw base elements.
* **Function Creation**: Refactored repeated patterns into reusable functions.
* **Refinement**: Adjusted colours, spacing, and size to match the original as closely as possible.

## Experiments and Outcomes

### Figure 1: Four Squares

|  |  |  |  |
| --- | --- | --- | --- |
| **Target**  **Image:** |  | **My**  **Output:** | **XXXXX** |

This figure took me XXX minutes to complete.

I used Generative AI tool to generate the initial draft of the figure. Some of the prompts I gave the tool was as follows:

* “*Draw the Microsoft logo*”
* etc.

The most difficult part of getting the diagram right was achieving equal spacing.

Initially, I drew four separate squares using copy-paste. However, I refactored this by creating a draw\_square() function and used a loop to move the turtle between squares. The hardest part was achieving equal spacing. I experimented with the penup() and goto() methods until alignment matched the figure.

### Figure 2: Interlocked Circles

|  |  |  |  |
| --- | --- | --- | --- |
| **Target**  **Image:** |  | **My**  **Output:** | **XXXXX** |

This figure took me XXX minutes to complete.

I used Generative AI tool to generate the initial draft of the figure. Some of the prompts I gave the tool was as follows:

* “*Draw the Olympics Rings*”
* etc.

The most difficult part of getting the diagram right was the thickness of the rings.

This figure taught me about relative positioning and layering. I used t.circle(50) and experimented with offsets to position the circles. Matching the overlapping effect required reordering draw calls and playing with turtle's heading to fine-tune vertical offset. The hardest part of the process was the thickness of the rings.

### Figure 3: Alien Character

|  |  |  |  |
| --- | --- | --- | --- |
| **Target**  **Image:** |  | **My**  **Output:** | **XXXXX** |

This figure took me XXX minutes to complete.

I used Generative AI tool to generate the initial draft of the figure. Some of the prompts I gave the tool was as follows:

* “*Draw a Space Invaders alien*”
* etc.

The most difficult part of getting the diagram right was the shape of the alien.

This was the most complex. I broke it into body parts: head, eyes, antennae, and body. I defined each part as a function and layered them. I faced challenges with symmetry and resolving overlaps.

### Figure 4: Shamrock

|  |  |  |  |
| --- | --- | --- | --- |
| **Target**  **Image:** |  | **My**  **Output:** | **XXXXX** |

This figure took me XXX minutes to complete.

I used Generative AI tool to generate the initial draft of the figure. Some of the prompts I gave the tool was as follows:

* “*Draw a shamrock*”
* etc.

The most difficult part of getting the diagram right was the shape of the stem.

Here, the trickiest bit was creating an equilateral triangle from three circles. I calculated points using angles (120° apart) and radius-based offsets. I also drew a stem using lines and a triangle base. Trial and error were used to ensure correct proportions.

## Reflection, Challenges & Learnings

### Challenges

* **Precise Alignment**: Especially in figures 1 and 4, getting the turtle to the correct location between drawings was non-trivial.
* **Colour Matching**: Since colour samples weren’t provided, I experimented using RGB values and Turtle’s colour names.
* **Overlapping Shapes**: Order of drawing mattered to create correct visuals (e.g., Olympic rings and alien eyes).

### Conclusion

This assignment improved my ability to think in shapes and translate ideas into code. It also strengthened my foundational Python skills and introduced me to the fun side of creative programming. I believe my diagrams closely match the originals and are backed by clean and structured code.

## Appendix A: My Code

### A1. Squares Code

|  |
| --- |
| Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx |

### A2. Olympic Rings code

|  |
| --- |
| Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx |

### A3. Space Invader alien code

|  |
| --- |
| Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx |

### A4. Shamrock code

|  |
| --- |
| Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx |