DS2030 Data Structures and Algorithms for Data Science
Lab 1 (Take Home) Due on August 12, 11.59pm

Instructions

- You are to use Python as the programming language. You may use VisualStudio Code (or any other editor you are comfortable with) as the IDE.
- You have to work individually for this lab.
- You are not allowed to share code with your classmates nor allowed to use code from the internet. You are encouraged to engage in high-level discussions with your classmates; however, ensure to include their names in the report/code documentation. If you refer to any source on the Internet, include the corresponding citation in the report/code documentation. If we find that you have copied code from your classmate or from the Internet, you will get a straight fail grade in the course.
- The submission must be a zip file with the following naming convention - rollnumber.zip. The Python files should be contained in a folder named after the question number.
- Include appropriate comments to document the code. Include a `read me` file containing the instructions for executing the code. The code should run on institute Linux machines.
- Upload your submission to Moodle by the due date and time. Do not email the submission to the instructor or the TA.

This lab will improve your understanding of linked lists.

1 Reordering Blueprints (5 points)

In a thrilling twist, Robin, a top-notch detective, has discovered a secret compartment in the Batcave containing a series of blueprints for advanced gadgets. These blueprints are critical for understanding Batman's most ingenious inventions. Each blueprint is represented as a node in a singly linked list.
To decode the blueprints and unveil the complete design of Batman's ultimate gadget, Robin must reorder the blueprints according to a special pattern devised by Batman. The pattern is essential for revealing the hidden functionality of the gadget.
Task 1: The reordering pattern is:

1. Start with the first blueprint.
2. Follow it with the last blueprint.
3. Then the second blueprint.
4. Followed by the second-last blueprint.
5. Continue this alternating pattern until all blueprints are reordered.

Task: Given the head of a singly linked list where each node represents a blueprint, reorder the list to follow this pattern.
Input Example:

```
B1 -> B2 -> B3 -> B4 -> B5
```

Output Example:

```
B1 -> B5 -> B2 -> B4 -> B3
```

Task 2: Batman, while deciphering the Joker's diabolical puzzles, stumbles upon a peculiar sequence involving numbers. The Joker challenges Batman to find out if a given number is a "happy number," which reveals a crucial piece of information about the Joker's next move, or a "sad number," which leads Batman into a trap. Batman needs to determine if a given number is a happy number to thwart the Joker's plans.
A number is considered a "happy number" if repeatedly summing the squares of its digits eventually leads to 1. If it falls into a cycle and never reaches 1, it is a "sad number."
Task: Given an integer n, return `true` if n is a happy number, and `false` if not.
Example 1:
Input:

```
n = 19
```

Output:

```
true
```

Explanation:

```
19 is a happy number because the process yields:
1^2 + 9^2 = 82
8^2 + 2^2 = 68
6^2 + 8^2 = 100
1^2 + 0^2 + 0^2 = 1
```

Example 2:
Input:

```
n = 2
```

Output:

```
false
```

Explanation:

```
2 is not a happy number because it ends up in a cycle:
2^2 = 4
4^2 = 16
1^2 + 6^2 = 37
3^2 + 7^2 = 58
5^2 + 8^2 = 89
8^2 + 9^2 = 145
1^2 + 4^2 + 5^2 = 42
4^2 + 2^2 = 20
2^2 + 0^2 = 4 (cycle repeats)
```

Constraints:

- $1 \leq n \leq 2^{31} - 1$

2 References

1. M Goodrich, R Tamassia, and M. Goldwasser, "Data Structures and Algorithms in Python", 1^{st} edition, Wiley, 2013.