111B Data Science and Python Programming

Homework Assignment #3

Due: 4/6 12:00:00

Once upon a time, a Flying Spaghetti Monster named UnknownEnzyme was pissed by those slackers during a group report in the Photonics Laboratory. After two terrible semesters with those suckers, she angrily buried her painstaking homework solution in the deep of an underground labyrinth and cursed all those people who came to find the solution. But things are not always satisfactory, otherwise we will not have this homework. One of her lovely teammates, Lightening, stalked her into this underground labyrinth, stole the solution, and became an inhabitant of RenÁi Di Bao. Since then, thousands of idiots have come to challenge this dungeon, trying to find the treasure and acquire this eastern mysterious power. However, the master of this dungeon, UnknownEnzyme, is not a vegetarian...

In front of the dungeon stood a giant pillar, bearing some inscriptions.

You need to download "HW3.zip" from E3 to accomplish this challenge.

- Lightening

!!!NOTICE!!!

In this homework, you are only allowed to import "**numpy**",
"**argparse**", built-in modules, and your own modules.

Problem #1: Build Dungeon Graph (40%)

The "**dungeons**" folder consists of multiple "**dungeon_{level}.txt**" files, which contain information about connected nodes and node content. In "**obj.py**" file, you will see several classes. The '***Node***' class object holds node name and node content, the "***Edge***" class stores source node and destination node, and the '***Dungeon***' class object uses "**dungeon_{level}.txt**" file as the input to generate a graph. Once the graph is built, print it out via '***__str__***' method.

The printed information must include:

1. Node name

2. Node content

3. Connected node name

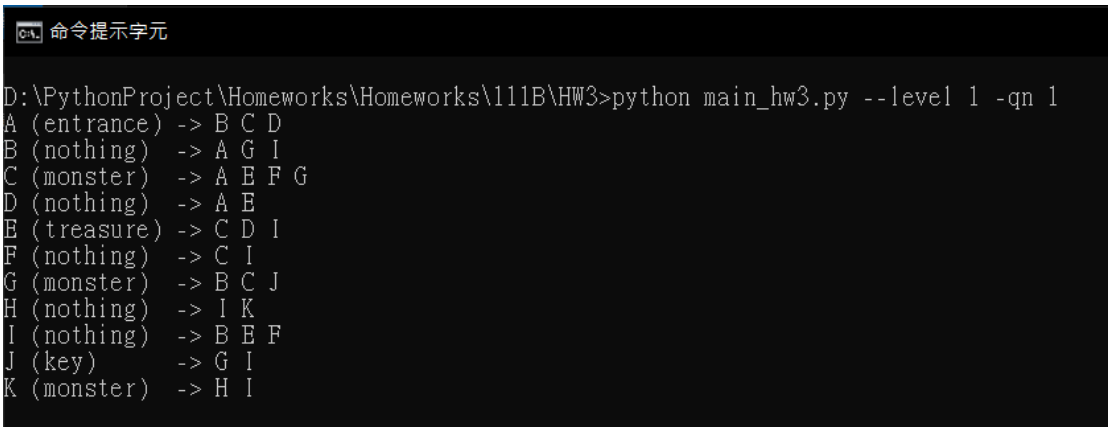

Fig.1 Printed information example.

For reading the text file in Python, you can use following code as reference:

```
with open('{path}/file.txt', 'r') as file:
    lines = file.readlines()
```

Problem #2: Cursed Dungeon (40%)

Their primary objective is to find the path leading to the key within the dungeon. Once they obtain the key, the challengers can proceed to the node containing the treasure. However, the dungeon is cursed with eastern mysterious power. Their mind will be confounded and they always lose their direction. Therefore, they can only go randomly until the treasure is found when they are holding a key. As the challengers wandering in this dungeon, they may encounter some monsters that causing irreversible damage. Your task is to count the number of challengers who successfully conquer the dungeon at different levels.

In "**obj.py**" file, the '***Challenger***' class object takes two input '***hp***' and '***dungeon***'. The parameter '***hp***' is the initial health point that the challenger has. After the challenger encounters a monster, the health point of the challenger will decrease by 1. The parameter '***dungeon***' receives '***Dungeon***' class object. Please complete '***random_walk***' method in '***Challenger***' class object with following illustration:

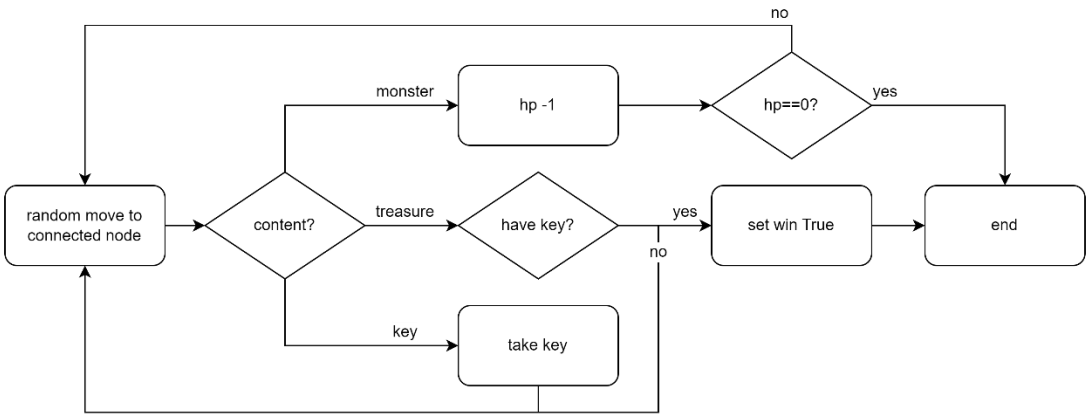

Fig.2 Workflow illustration of '***random_walk***' method.

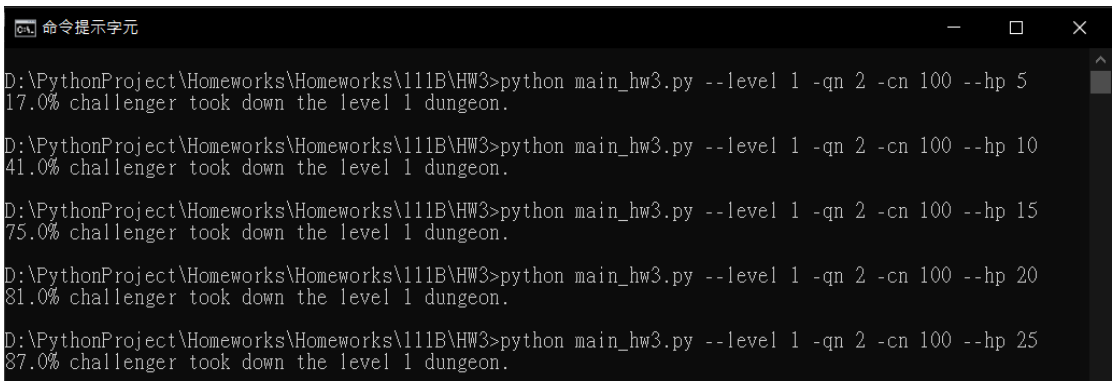

Fig.3 Problem #2 result with different '***hp***'.

Problem #3: Dispelling the curse (20%)

Among these challengers are some powerful wizards who can use their magic to alleviate the curse of the dungeon. This magic increases the probability of specific node being selected. As a result, they can identify the shortest path nodes leading to the key or treasure and increase the probability of those nodes (They are cheating!!!). Your task is to count the number of challengers who successfully challenge the dungeon at different probabilities.

The probability of nodes being selected from current node will be rescaled by following equation:

$$p(n_{other}) = \frac{1}{N}(1-\alpha), \qquad p(n_{selected}) = 1 - \frac{N-1}{N}(1-\alpha)$$

N is the number of connected nodes

Please complete '***weighted_random_walk***' method, which takes '***alpha***' as input, in '***Challenger***' class object with following illustration:

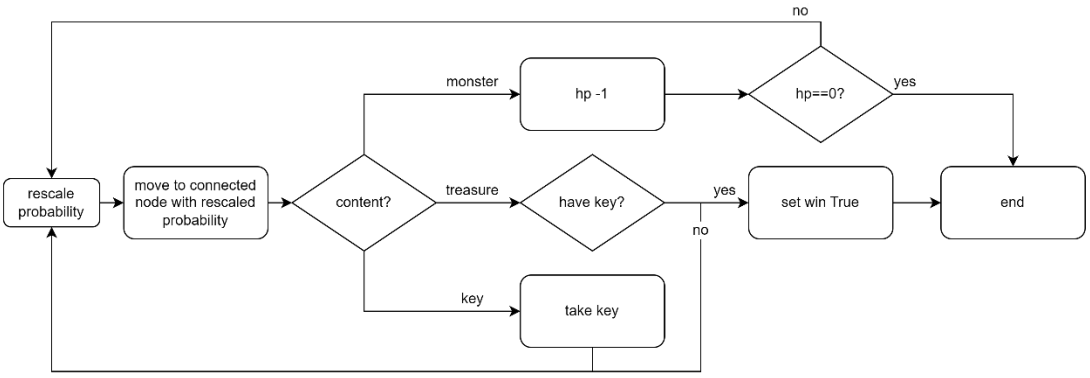

Fig.4 Workflow illustration of '***weighted_random_walk***' method.

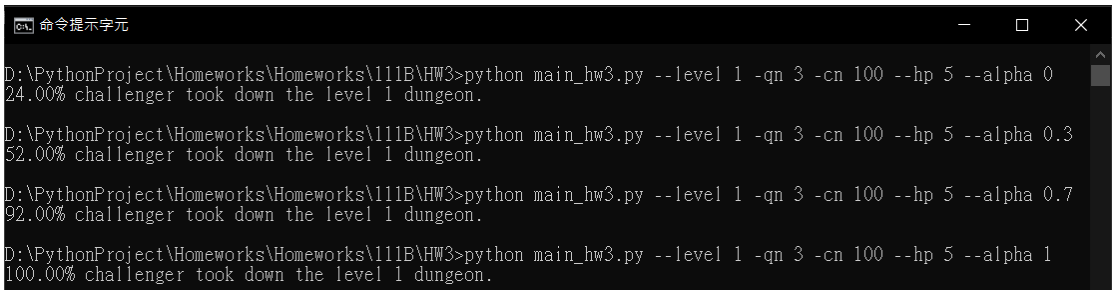

Fig.5 Problem#3 result with different '***alpha***'

Please accomplish this homework with an organized code (e.g., with main script and function script). For example, you can package your scripts that related to the class object in a module "**obj.py**", some useful functions in other module, and remain the main content in the main script "**main_hw3.py**" clear. In addition, you should use "**argparse**" to set all related parameters of this homework. Here is a template for your code structure:

```
111B_hw3_0123456789
    ├─ dungeons          # folder contains dungeon.txt
    ├─ obj.py            # Objects
    └─ main_hw3.py       # Main scripts of hw3
```

You don't need to follow this structure, just keep your main script clean.

Hand in procedure:

As we had mentioned in the lecture, you should list all your collaborators in your programs. Here is the template:

```
"""
Created on Sun Aug 7 01:23:45 2022

@author: Xi Winnie, student ID

@collaborators: Jane Doe, her student ID
                John Doe, his student ID
"""
```

Please save your code as a "**.zip**", "**.7z**", or "**.rar**" file, where the file name should follow this format:

111B_hw3_ID.zip

For example,

111B_hw3_0123456789.zip

Please be aware. **We are not going to accept any homework file with wrong file name or without signature**. Please double check the content of your files.

Once you have accomplished your works, you can upload your homework to the "E3@NYCU" system. There will be a section for uploading your homework.