

Game Detector

Instructions

- Choose a language among Ruby, Python and Javascript.
- Implement what is requested.
- Write tests to validate your implementation.
- Send your work in a compressed archive containing implementation file, test file, and whatever comments/notes helped you solve the problem.
- Please complete your work on time and do not worry if you can't get everything perfect.

Input

- A mapping from a game id to a list of possible n-grams used to refer to the game, for example:

```
{
  "CallOfDutyWW2": ["Call of duty world war two", "COD WW2", "COD WWII", "WW2COD"],
  "Fortnite": ["Fortnite", "Fort Nite"],
  "Destiny": ["Destiny", "original Destiny game"],
  "Destiny2": ["Destiny 2", "the last Destiny game", "Destiny II"],
  "WorldOfWarcraft": ["WoW the game", "world of warcraft"]
}
```

- A large body of documents in the form of an array of strings, for example:

```
["I liked the last Destiny game, now I play Fortnite",
 "Lol, no comment about that",
 ...,
 "I'm still playing world of warcraft since ww2"]
```

Output

The documents with tagged games (*see example*)

```
["I liked TAG{Destiny2,the last Destiny game}, now I play TAG{Fortnite,Fortnite}",
 "Lol, no comment about that",
 ...,
 "I'm still playing TAG{WorldOfWarcraft,world of warcraft} since ww2"]
```

Note

A recognized sequence of words, e.g. *“Call of Duty world war two”*, identified to refer to a certain game id, e.g. *“CallOfDutyWW2”*, must be represented in the final text as *TAG{GameID,original text}*, e.g. *TAG{CallOfDutyWW2,Call of Duty world war two}*.

Extra points

Comment on the running time of your implementation.

Have fun!