Assignment 1

COMP9021, Term 3, 2023

1. General matters

1.1. **Aims.** The purpose of the assignment is to:

- let you design a solution to a problem that requires to parse and analyse text and perform logical computations;
- let you implement this solution in the form of a short Python program;
- practice reading from a file and making use of, in particular, tests, repetitions, lists, sets, strings, Boolean operators.

1.2. **Submission.** Your program will be stored in a file named `knights_and_knaves.py`. After you have developed and tested your program, upload it using Ed (unless you worked directly in Ed). Assignments can be submitted more than once; the last version is marked. Your assignment is due by October 23, 10:00am.

1.3. **Assessment.** The assignment is worth 13 marks. It is going to be tested against a number of input files. For each test, the automarking script will let your program run for 30 seconds.

Assignments can be submitted up to 5 days after the deadline. The maximum mark obtainable reduces by 5% per full late day, for up to 5 days. Thus if students A and B hand in assignments worth 12 and 11, both two days late (that is, more than 24 hours late and no more than 48 hours late), then the maximum mark obtainable is 11.7, so A gets $\min(12.7, 11) = 11$ and B gets $\min(11.7, 11) = 11$.

The outputs of your programs should be **exactly** as indicated.

1.4. **Reminder on plagiarism policy.** You are permitted, indeed encouraged, to discuss ways to solve the assignment with other people. Such discussions must be in terms of algorithms, not code. But you must implement the solution on your own. Submissions are routinely scanned for similarities that occur when students copy and modify other people's work, or work very closely together on a single implementation. Severe penalties apply.

Raymond Smullyan has designed many puzzles involving Knights and Knaves. Knights always tell the truth, whereas Knaves always lie. We refer to Knights and Knaves as Sirs. A puzzle, which is a set of English sentences, involves a finite number of Sirs. Solving the puzzle means:

- determining the names of all Sirs involved in the puzzle;
- determining solutions to the puzzle, where a solution qualifies each Sir as either a Knight or a Knave.

Some puzzles have no solution, others have a unique solution, and others have at least 2 solutions. The following is an example of a puzzle with a unique solution.

> `One evening as you are out for a stroll, you walk by a doorway labeled no normals allowed. Some people are talking inside. Curious, you listen, and you hear Sir Paul who says: "all of us are Knaves." "Exactly one of us is a Knight," replies Sir Jenny. As for Sir John, who is also inside, he just keeps quiet. Who is a Knight, and who is a Knave?`

The Sirs involved in this puzzle are Sir Jenny, Sir John, and Sir Paul. The unique solution is given by Sir Jenny being a Knight, Sir John being a Knave, and Sir Paul being a Knave.

2. Detailed description

2.1. **Syntax of puzzles.** A sentence starts with a capital letter and ends in a full stop, an exclamation mark, or a question mark, possibly followed by closing double quotes. Sir, Sirs, Sir names, Knight and Knave always start with a capital letter, and no other word inside a sentence is capitalised. A sentence in a puzzle contains at most one part enclosed between double quotes. When a sentence contains one part enclosed between double quotes, the part outside the double quotes contains a single occurrence of the form `Sir` *`Sir_Name`*, and what occurs between the double quotes is something said by `Sir` *`Sir_Name`*. A sentence that contains no part enclosed between double quotes might refer to a number of Sirs, always in the form `Sir` *`Sir_Name`*, or `Sirs` *`Sir_Name_1`* `and` *`Sir_Name_2`*, or `Sirs` *`Sir_Name_1`*`,` *`Sir_Name_2`*`,` …`and` *`Sir_Name_n`*, where $n \geq 3$, and *`Sir_Name_1`*, …, *`Sir_Name_n`* are pairwise distinct.

What is between double quotes is a sentence in one of the following forms, ending in either a comma, a full stop, an exclamation mark, or a question mark:

- `At/at least one of` *`Conjunction_of_Sirs`*`/us is a Knight/Knave`
- `At/at most one of` *`Conjunction_of_Sirs`*`/us is a Knight/Knave`
- `Exactly/exactly one of` *`Conjunction_of_Sirs`*`/us is a Knight/Knave`
- `All/all of us are Knights/Knaves`
- `I am a Knight/Knave`
- `Sir` *`Sir_Name`* `is a Knight/Knave`
- *`Disjunction_of_Sirs`* `is a Knight/Knave`
- *`Conjunction_of_Sirs`* `are Knights/Knaves`

where:

- *`Disjunction_of_Sirs`* is in one of the following forms:
  - *`Sir_1`* `or` *`Sir_2`*
  - *`Sir_1`*`,` *`Sir_2`*`,` … `or` *`Sir_n`* ($n \geq 3$)
- *`Conjunction_of_Sirs`* is in one of the following forms:
  - *`Sir_1`* `and` *`Sir_2`*
  - *`Sir_1`*, *`Sir_2`*`,` … `and` *`Sir_n`* ($n \geq 3$)
- *`Sir_1`*, …, *`Sir_n`* are pairwise distinct expressions of the form `Sir` *`Sir_Name`* or `I`.

2.2. **Input and output of programs.** Your program will prompt the user for a text file, assumed to be stored in the working directory, that stores the sentences that make up a puzzle. No assumption should be made on the number of English sentences provided as input, nor on the length of a sentence, nor on the length of a Sir name, nor on the number of Sirs involved in the puzzle.

Your program should:

- output the Sirs involved in the puzzle in lexicographic order (5 marks);
- output whether or not there is a solution, and in case there is, how many there are (3 marks);
- in case a unique solution exists, output it, with all Sirs being qualified as either Knight or Knave in alphabetical order (5 marks).

2.3. **Sample outputs.** Here are a few tests together with the expected outputs. The outputs of your program should be exactly in accordance with the following outputs. Outputs of your program will be matched against expected outputs line for line.

```
$ cat test_1.txt
I have just seen Sirs Sanjay and Eleonore!
"I am a Knave," whispered Sir Eleonore.
Who is a Knight and who is a Knave?
$ python3 knights_and_knaves.py
Which text file do you want to use for the puzzle? test_1.txt
The Sirs are: Eleonore Sanjay
There is no solution.

$ cat test_2.txt
I have just met Sirs Frank, Paul and Nina.
Sir Nina said: "I am a Knight," but I am not sure
if that is true. What do you think?
$ python3 knights_and_knaves.py
Which text file do you want to use for the puzzle? test_2.txt
The Sirs are: Frank Nina Paul
There are 8 solutions.

$ cat test_3.txt
Yesterday, I visited Sirs Andrew and Nancy. I asked Sir Andrew
who he was, and he answered impatiently: "Sir Nancy and I
are Knaves!" Then I met Sir Bill who introduced me to his wife
and told me: "at least one of Sir Hilary
and I is a Knave." Should I trust them?
$ python3 knights_and_knaves.py
Which text file do you want to use for the puzzle? test_3.txt
The Sirs are: Andrew Bill Hilary Nancy
There is a unique solution:
Sir Andrew is a Knave.
Sir Bill is a Knight.
Sir Hilary is a Knave.
Sir Nancy is a Knight.
```