

Mini-Project 5: Monster Diagnosis (Fall 2022)

In this project, you'll implement an agent that can diagnose monster diseases. Based on a list of diseases and their ailments and a list of elevated and reduced vitamin levels, you will diagnosis the disease(s) affecting a particular monster. You will submit the code for diagnosing these monsters to the Mini-Project 5 assignment in Gradescope. You will also submit a report describing your agent to Canvas. Your grade will be based on a combination of your report (50%) and your agent's performance (50%).

About the Project

Monster physiology relies on a balance of 26 vitamins, conveniently named Vitamin A through Vitamin Z. Different monster ailments can cause elevated or reduced levels for different vitamins. Unfortunately, every ailment affects every monster species differently, so there is no canonical list of all monster ailments and their effects; instead, the ailments must be interpreted in the context of a particular species.

In this project, you'll diagnose a particular monster based on its symptoms and a list of ailments and their effects on that monster species. Both the symptoms of an ailment and a monster's symptoms will be represented by a dictionary, where the keys are the different vitamin letters and the values are either + (for elevated), - (for reduced), or 0 (for normal). For example:

```
{"A": "+", "B": "0", "C": "-", "D": "0", "E": 0, "F": "+", ...}
```

This would represent elevated levels of Vitamins A and F, and a reduced level of Vitamin C. This could be the symptoms of a particular patient (e.g. "Sully presented with elevated levels of Vitamins A and F, and a reduced level of Vitamin C), or it could be the symptoms of a particular disease (e.g. "Alphaitis causes elevated Vitamins A and F and a reduced Vitamin C in monsters like Sully").

Your Agent

To write your agent, download the starter code below. Complete the `solve()` method, then upload it to Gradescope to test it against the autograder. Before the deadline, make sure to select your best performance in Gradescope as your submission to be graded.

Starter Code

Here is your starter code: [MonsterDiagnosisAgent.zip](#). You may also access it at the [course's Github repository](#).

The starter code contains two files: `MonsterDiagnosisAgent.py` and `main.py`. You will write your agent in `MonsterDiagnosisAgent.py`. You may test your agent by running `main.py`. You will only submit `MonsterDiagnosisAgent.py`; you may modify `main.py` to test your agent with different inputs.

Your `solve()` method will have two parameters. The first will be a list of diseases and their symptoms. This will be provided as a dictionary, where the keys are the names of diseases and their values are each a dictionary representing its symptoms. For example:

```
{"Alphaitis": {"A": "+", "B": "0", "C": "-", "D": "0", "E": "0", "F": "+"}, ...}, {"Betatosis": {"A": "0", "B": "+", "C": "-", "D": "0", "E": "+"}, {"F": "-", ...}}, {"Gammanoma": {"A": "0", "B": "0", "C": "+", "D": "+", "E": "+", "F": "+", ...}, ...}]
```

There may be up to 24 diseases. Each disease will have values for all 26 vitamins. Most vitamins will be unaffected by any particular disease; most diseases only affect 3-6 vitamins.

The second parameter to the function will be a particular set of symptoms, given as a dictionary, such as:

```
{"A": "+", "B": "0", "C": "-", "D": "0", "E": "0", "F": "+", ...}
```

Your goal is to identify the smallest subset of diseases from the list of ailments that can explain the monster's symptoms.

If the patient has two diseases with opposite effects, they cancel each other out. For example, if a patient had both Alphaitis and Betatosis (according to the definitions above), they would have a normal level of Vitamin F because Alphaitis elevates F and Betatosis reduces F.

If the patient has two diseases with the same effect, their effect remains the same. For example, if a patient had both Alphaitis and Betatosis (according to the definitions above), they would have a reduced level of Vitamin C because both diseases reduce Vitamin C. There is no extra effect from having multiple diseases with the same effect.

If a patient has more than two diseases, then each Vitamin moves in whichever direction is caused by the largest number of diseases. For example, if a patient had Alphaitis, Betatosis, and Gammanoma, they would exhibit reduced levels of Vitamin C: both Alphaitis and Betatosis reduce Vitamin C, while Gammanoma elevates it. Two reductions plus one elevation leads to a reduction. If, on the other hand, they had four diseases, two

of which reduced Vitamin C and two of which elevated Vitamin C, their Vitamin C levels would be normal.

Returning Your Solution

Your `solve()` method should return a list of strings. Each string should be the name of one of the (up to) 24 diseases. Together, the diseases should explain all of the symptoms of the patient. If there are multiple sets of diseases that can explain all the symptoms, then you should return the set with the minimum number of diseases according to the principle of parsimony. For this project, you may assume that all diseases are equally likely and that all symptoms will be covered.

For the two test cases in the starter code, the answers should be: ["Alphaitis", "Betatosis"] and ["Gammanoma", "Deltaccol", "Episicusus"].

Submitting Your Solution

To submit your agent, go to the course in Canvas and click Gradescope on the left side. Then, select CS7637 if need be.

You will see an assignment named Mini-Project 5. Select this project, then drag your `MonsterDiagnosisAgent.py` file into the autograder. If you have multiple files, add them to a zip file and drag that zip file into the autograder.

When your submission is done running, you'll see your results.

How You Will Be Graded

Your agent will run against 20 test cases. The first four of these will always be the same; these are those contained in the original `main.py`. The last 16 will be randomly generated.

You can earn up to 40 points. You will earn one point for each of the test cases for which you correctly identify a list of diseases that explain all the symptoms. You will earn an additional point for each of the test cases for which you identify the *smallest* list of diseases that explain all the symptoms.

You may submit up to 40 times prior to the deadline. The large majority of students do not need nearly that many submissions, so do not feel like you should use all 40; this cap is in place primarily to prevent brute force methods for farming information about patterns in hidden test cases or submitting highly random agents hoping for a lucky submission. Note that Gradescope has no way for us to increase your individual number of submissions, so we cannot return submissions to you in the case of errors or other issues, but you should have more than enough submissions to handle errors if they arise.

You **must** select which of your submissions you want to count for a grade prior to the deadline. Note that by default, Gradescope marks your last submission as your

submission to be graded. We cannot automatically select your best submission. Your agent score is worth 50% of your overall mini-project grade.

Your Report

In addition to submitting your agent to Gradescope, you should also write up a short report describing your agent's design and performance. Your report may be up to 4 pages, and should answer the following questions:

- How does your agent work? Does it use some concepts covered in our course? Or some other approach?
- How well does your agent perform? Does it struggle on any particular cases?
- How efficient is your agent? How does its performance change as the number of diseases grows?
- Does your agent do anything particularly clever to try to arrive at an answer more efficiently?
- How does your agent compare to a human? Do you feel people approach the problem similarly?

Assignment Project Exam Help

You are encouraged but not required to include visuals and diagrams in your four page report. The primary goal of the report is to share with your classmates your approach, and to let you see your classmates' approaches. You may include code snippets if you think they are particularly novel, but please do not include the entirety of your code.

WeChat: cstutorcs

Tip: Remember, we want to see how you put the content of this class into action when designing your agent. You don't need to use the principles and methods from the lectures precisely, but we want to see your knowledge of the content reflected in your terminology and your reflection.

Submission Instructions

Complete your assignment [using JDF](#), then save your submission as a PDF. Assignments should be submitted to the corresponding assignment submission page in [Canvas](#). You should submit a **single** PDF for this assignment. This PDF will be ported over to Peer Feedback for peer review by your classmates. If your assignment involves things (like videos, working prototypes, etc.) that cannot be provided in PDF, you should provide them separately (through OneDrive, Google Drive, Dropbox, etc.) and submit a PDF that links to or otherwise describes how to access that material.

This is an individual assignment. All work you submit should be your own. Make sure to cite any sources you reference, and use quotes and in-line citations to mark any direct quotes.

Late work is not accepted without advanced agreement except in cases of medical or family emergencies. In the case of such an emergency, please [contact the Dean of Students](#).

Grading Information

Your report is worth 50% of your mini-project grade. As such, your report will be graded on a 40-point scale coinciding with a rubric designed to mirror the questions above. Make sure to answer those questions; if any of the questions are irrelevant to the design of your agent, explain why.

Peer Review

After submission, your assignment will be ported to [Peer Feedback](#) for review by your classmates. Grading is *not* the primary function of this peer review process; the primary function is simply to give you the opportunity to read and comment on your classmates' ideas, and receive additional feedback on your own. All grades will come from the graders alone. See the course [participation policy](#) for full details about how points are awarded for completing peer reviews.

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs