|  |
| --- |
| **Artificial Intelligence - Lab (AL2002)** |
| **Course Instructor(s):** |
| Mr. Ahsan Shakeel Malik, Ms. Nabeelah Maryam  **Section(s): BCY-A,BCY-B** |

|  |  |
| --- | --- |
| **Sessional-I Exam** | |
| **Total Time (Hrs):** | **1** |
| **Total Marks:** | **30** |
| **Total Questions:**  **Date:** Mar 17, 2025 | **2** |

**\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Roll No Course Section Student Signature**

**Do not write below this line.**

**Attempt all the questions.**

**Instructions:**

1. If you are late to the exam no additional time will be provided.
2. Download the 2 skeleton code notebooks provided to you on GCR.
3. Run all the cells on the skeleton code notebook before writing your implementation.
4. There are a total of 2 questions. Attempt both of them. Read them carefully, understand the question, and then attempt them.
5. No additional sheet will be provided for rough work. Use the back of the last page for rough work.
6. Calculator sharing is strictly prohibited.
7. Dataset and **Python Notebook** are already provided to you on GCR.
8. Write your submission time on the paper before you submit it to the invigilator
9. If plagiarism is found in your code, **F grade** will be awarded.
10. You are not allowed to use any **AI Tool/GPT or Internet** for this exam. If caught your exam will be cancelled and will be awarded 0 marks or F grade in the course.
11. Anyone caught cheating during the exam will get a **0** in the exam and might face a DC inquiry. Please remember that you are not allowed to discuss the exam with your peers within the confines of the examination hall
12. **Submissions will be opened 15 minutes** before the end of the exam. Make sure to follow the naming convention instructions given at the end. Failure to comply will be result in mark deductions. **Your total time for attempting is 1 hours 30 mins** the last 15 mins are only for submissions.
13. **Naming Convention for Submission: <*RollNo\_Sec\_MID\_Part1.ipynb>* <*RollNo\_Sec\_MID\_Part2.ipynb>***
14. **No Submissions will be accepted after the 15 minutes allotted for submission.**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Q-1** | **Q-2** | **Total** |
| **Marks Obtained** |  |  |  |
| **Total**  **Marks** | **15** | **15** | **30** |

**[CLO 2: Design and Develop intelligent programs using python.]**

**Q1:** Using **numpy and pandas** libraries perform the following operation on the dataset.  **[15 marks]**

The operations are mentioned in the notebook provided. **You must write each operation in its cell. Do not combine all operations in a single cell.**

**[CLO 3: Demonstrate Knowledge of blind as well as informed search and ability to practically apply the corresponding techniques]**

**Q2:** In this question you are required to implement a **Word Ladder** using **A\* Search [15 marks]**

**Algorithm.** The goal here is to reach a target word from the start word in the ladder. Each intermediate step consist of performing a transformation where a single character is changed in the word.

For Example: cat -> hat -> hot -> lot -> log -> dog

Where **Start Word = cat**  and **Target Word = dog**.

**Problem Technical Explanation:**

You are given a word file “**words.txt**”. This file contains a list of English language words.

In your skeleton code file. This words.txt is being read and being used to create a **graph** variable via **build\_graph function**. This graph variable is a **dictionary** with words as key and a list of its neighbors.

For example: { ‘cat’ : [‘bat’ ,‘hat’, ‘lat’] } , here ***cat*** is *key* and **bat,hat,lat** are its neighbors.

The neighbors of a word are created by performing a one letter change in the original word.

In the skeleton code you are also given a **heuristic function** which is used to compute heuristics value of a word to the target word. This is computed by checking how many letters are different between the target word and our current word.  
 For example: heuristic value between **cat** and **hat** is 1 because only 1 character is different. And heuristic value between **cat** and **dog** is 3 because all values are different.

**Assumption for This Problem:**

As we are using A\* search so we need cost between nodes as well or cost to neighbor as well.

So in this scenario you can assume the cost between a node and its neighbor is always **1.**

For example: In our graph the word “*cat*” has neighbors “*bat,hat,lat*” so cost of ***cat*** to each of its neighbor is **1**.

**Your Implementation:**

You are required to implement an A\* search algorithm in the cell where its function is defined. You are given a graph. A function to compute heuristics between a word/node to target word/node. There is also a *is\_valid\_word()* function that is used to check whether a word exists in the dictionary.