|  |  |
| --- | --- |
| **Exp. No.** | **Experiment Title** |
| 1 | WAP to implement DFS and BFS for traversing a graph from source node (S) to goal node (G), where source node and goal node is given by the user as an input. |
| 2 | Design water jug problem solver  You are given two jugs with m litres and a n litre capacity. Both the jugs are initially empty. The jugs don’t have markings to allow measuring smaller quantities. You have to use the jugs to measure d litres of water where d is less than n. You are given two jugs with m litres and a n litre capacity. Both the jugs are initially empty. The jugs don’t have markings to allow measuring smaller quantities. You have to use the jugs to measure d litres of water where d is less than n.  rubrics  can explain the algorithm implemented with justification of choice of algorithm 5 marks  user input can be dynamic 1 marks  execution completes with proper output all cases 2 marks  Lab discipline 2 marks  conscious use of internet is allowed |
| 3 | Solve 8 puzzle problem using A\* algorithm where initial state and Goal state will be given by the users.  Those who solved water jug using A\* have to apply AO\*  Rubrics:  Proper Comments about the algorithm steps 4 marks  dynamic input 2 marks  Language used other than python 1 mark  No use of internet 3 marks  Use of internet -3 marks |
| 4 | Implement using C/C++, the Fixed Increment Perceptron Learning algorithm as presented in the attachment.  The training set for a 2- classificaiton problem is also attached (check Teams). Iterate the perceptron through the training set and obtain the weights. |
| 5 | Given a c++ code bnp, identify the algorithm implemented through the code. Also document the code |
| 6a | Understand the project available on following link  Project Link: <https://github.com/aharley/nn_vis>  Project by: <https://adamharley.com/>  Reference in case needed: <https://www.youtube.com/watch?v=pj9-rr1wDhM> |
| 6b | Part 2  Populate the table below to summarize your understanding of the project mentioned in part 1   |  |  |  | | --- | --- | --- | | Layer | Task | Rationale | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
|  | How does the following hyper-parameters affect the network performance   |  |  |  | | --- | --- | --- | | **Hyper-Parameter** | **One Line Definition** | **Effect on the CNN** | | Stride |  |  | | Dilation Rate |  |  | | Type of pooling layer |  |  | | Kernel size |  |  | | padding |  |  | |
|  | References:  [An Intuitive Explanation of Convolutional Neural Networks – the data science blog (ujjwalkarn.me)](https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/)  [Gentle Dive into Math Behind Convolutional Neural Networks | by Piotr Skalski | Towards Data Science](https://towardsdatascience.com/gentle-dive-into-math-behind-convolutional-neural-networks-79a07dd44cf9)  [Intuitively Understanding Convolutions for Deep Learning | by Irhum Shafkat | Towards Data Science](https://towardsdatascience.com/intuitively-understanding-convolutions-for-deep-learning-1f6f42faee1)  [An Introduction to different Types of Convolutions in Deep Learning | by Paul-Louis Pröve | Towards Data Science](https://towardsdatascience.com/types-of-convolutions-in-deep-learning-717013397f4d) |
|  | Rubrics:  Part 1 (Indirect)  Part 2 Layer Task – 5 points  Hyper Parameter Task – 5 points |
| 7 | Prepare your version of CNN following the steps in the link shared here.  <https://towardsdatascience.com/build-your-own-convolution-neural-network-in-5-mins-4217c2cf964f> |
| 8 | Design a Convolutional Neural Network from Scratch for MNIST fashion dataset. Apply dropout technique to deal with the overfitting. Dataset can be downloaded from the Kaggle. |
| 9 | Design a Machine Learning model for the Heart Attack prediction. Download the Dataset from  <https://physionet.org/about/database/> along with the description |
| 10 | Design Machine Learning model for application of your choice utilizing Kaggle data for training. Apply genetic algorithm to optimize the performance of the model. |
| 11a | Understanding the basics and IDE for Prolog Programming |
| 11b | Implement any two of the following using Prolog: - Medical diagnosis of common cold and flu using symptom inputs   * Demonstrating list in prolog * Monkey banana problem * Find the factorial of a given number |
| 12 | WAP to design Tic Tac Toe games from O (Opponent) and X (Player) by using minimax algorithm. |
| 13 | Final Project |