CS5205: Advanced Artificial Intelligence Lab

Assignment - 1

16/01/2025

Let us consider a course work in some institute. The instructor has provided a set of assignments which need to be solved by the students. The assignments have some dependencies among themselves. Please see Figure 1. For example, Assignment 6 can be completed once Assignments 1 & 5 are completed. It means that one can proceed for an assignment only when its predecessors have been completed. In order to reduce the burden on students, the instructor has allowed to work in a group of N students but without the help of ChatGPT or any other LLMs. As the institute has a subscription for a pro-account for all students in one such well known LLM, students have decied to utilize such facility for solving their assignments. However, there is a limit on the number of prompts (K) that one student can send on a given day. A student cannot share his/her remaining prompts with an another student. A single assignemnt has to be done by one student only. It is known that the (exact) number of prompts needed to solve each of the assignments. For example, one needs 3 prompts to find out the solution for Assignment 4. Additionally, an assignment cannot be solved paritally on a given day. It means if a student does not have sufficient prompts available to solve an assignment on a day then he/she has to start afresh the same assignment on the next day. One can solve more than one assignment in a given day provided necessary prompts are available with him/her. Goal is to finish all assignments in m number of days, where m will be an input to be provided in the command line.

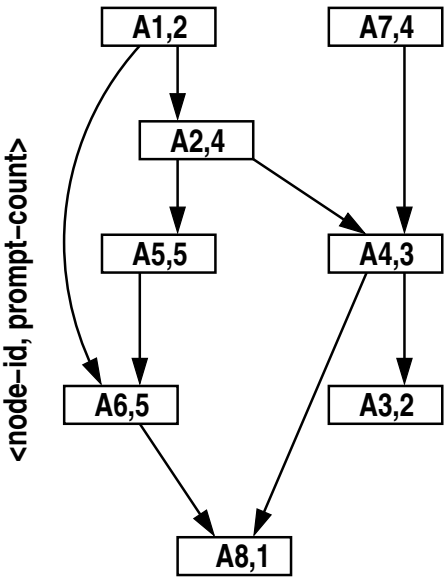

Figure 1: Sample scenario

Sample input:

```
% Comment line Ignore
% N = Number of students in a group
N 3

% K = Number of prompts allowed per day
K 5

% Assignment details with required prompts and Dependencies
% A <id> <prompt-count> <dependencies> <terminating-symbol>
A 1 2  0
A 2 4 1  0
A 3 2 4  0
A 4 3 7 2  0
A 5 5 2  0
A 6 5 5 1  0
A 7 4  0
A 8 1 6 4 0
```

To-do:

- Print all possible valid schedules.
- You need write program in C or C++ or Python.
- Command line interface should be as follows: <executable-name> <input-filename> <number-of-days>
- You need to submit your code along with 3 sample input files. Make sure that each input contains at least 10 assignments. The dependency structure in each file should be different.
- You need to submit a README.txt file that describes how to compile / run your program, any other details.