Assignment 2: Two probability based models

The goal of the assignment is to help student get familiar with two probability based models. It consist of two tasks: (1) to implement forward-backward algorithm (smoothing) and (2) to extract the Bayesian Network probability distribution from data.

Task 1: forward-backward algorithm.
In this task, you will work on Smoothing.java or Smoothing.py.

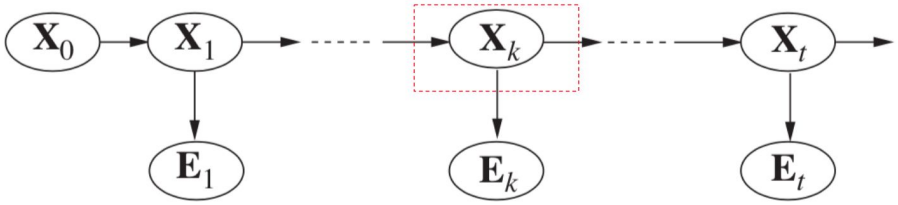
You will work on the **umbrella & weather task**, which have been discussed a lot in the class, and which is also explained clearly in the textbook. It says that a secret guard living in the underground installation, wants to know whether it's raining today by observing the director coming in the office with or without an umbrella.

It is a hidden markov model. Observed evidence variable E represents whether or not the director takes an umbrella. Hidden state variable X represents the weather, which can be raining or sunny.

Transition model	today rain	today sunny
yesterday rain	0.7	0.3
yesterday sunny	0.3	0.7

Sensor model	take umbrella	no umbrella
rain	0.9	0.1
sunny	0.2	0.8

You will implement the forward-backward algorithm, which is used to do the smoothing inference: compute the probability distribution of a past time state, given all evidence up to the present. As shown in **Figure 15.4 in textbook.**

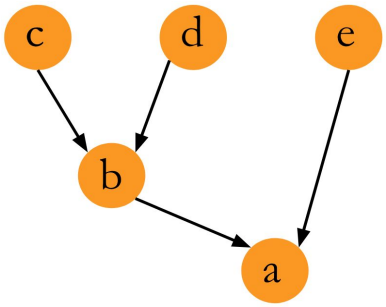


You are provided with 100 day of the director behavior data (with/without umbrella) in **assign2_umbrella.txt**. You need to produce the smoothing inference based weather probability distribution for day 1 to day 100.

Initialization: as shown in Figure 15.4 in textbook, you can initialize the backward message **b** as all "1".
Implementation: you need to fill the codes in **smoothing()**, so that main function can output correct smoothed weather probabilities.

Task 2: Extract the Bayesian Network probability distribution from data.
In this task, you will work on BayesianNetwork.java or BayesianNetwork.py.

For the given Bayesian Network structure, as shown below, please generate the probability distribution of each node from the given data in **assign2_BNdata.txt**.



Joint probability: P(a,b,c,d,e)=P(a|b,e)P(b|c,d)P(c)P(d)P(e)
a has 2 values
b has 3 values
c has 3 values
d has 2 values
e has 2 values

You need to implement **get_p_b_cd() and get_p_a_be()**.

Requirements and Reminders:

1. You **CANNOT** change the classes' names or the required methods' names. However, you can add new variables, constants, and methods in these classes and create new classes if necessary.
2. You are **NOT** allowed to change anything in **main function** for running your assignment 2 in two tasks.
3. You **CAN ONLY** use Java or Python in this assignment.
4. You **CANNOT** use external Java or Python packages.

Grading:
Your submission will be graded based on:

1. Correctness of the implementation on the required functions (70%)
2. Efficiency of your implementation, make sure your code finish processing two collections within 10 minutes (20%)

3. Necessary program annotation and commentaries (10%)

Submission Requirements

A zipped file package with the naming convention as “pittids_a2”. For example, suppose the Pitt id is jud1, then the submission package should be jud1_a2.zip.
The file package should contain:

1. All the scripts/programs you used for this assignment (**src folder**)
2. Your output in the screen. (This should in **txt file**.)