UNIVERSITY OF LINCOLN

Lincoln School of Computer Science

Assessment Item Briefing Document

Title: CMP9132M Advanced Artificial Intelligence **Assessment Item 1 (assignment)**	**Indicative Weighting: 50%**

Learning Outcomes:

On successful completion of this assessment item a student will have demonstrated competence in the following areas:

- [LO1] Critically appraise a range of AI techniques for decision-making, problem solving and learning, identifying their strengths and weaknesses, and selecting appropriate methods to serve particular roles;
- [LO3] Design and develop an AI-based software program for solving complex search problems in an application domain of interest.

Requirements

This assessment comprises two assessed Tasks, as detailed in the following page.

1. TASKS ON PROBABILITIES (Task 1.A and Task 1.B). Weighting: 50% of this component.

2. TASK ON MARKOV MODELS. Weighting: 50% of this component.

Your submission should include a concise report (maximum 5 pages, font size of 11 or 12 points, not including the cover sheet, references, and appendixes) that describes your work on the above tasks. Whenever possible, you should cite previous works from the related literature to backup your arguments or choices.

Useful Information

This assessment is an individual assessment component. Your work must be presented according to the Lincoln School of Computer Science guidelines for the presentation of assessed written work. Please make sure you have a clear understanding of the grading principles for this component as detailed in the accompanying Criterion Reference Grid.
If you are unsure about any aspect of this assessment component, please seek the advice of a member of the delivery team.

Submission Instructions

The deadline for submission of this work is included in the School Submission dates on Blackboard.

You must make an electronic submission of your work in PDF format by using the assessment link on Blackboard for this component. Each student is then required to discuss their solution to the module instructors, as per the schedule indicated on Blackboard. You must attend the lectures for further details, guidance and clarifications regarding these instructions.

DO NOT include this briefing document with your submission.

1. TASKS ON PROBABILITY

For each one of the following tasks, select and critically appraise the most relevant technique(s) to solve the problem, justify your choice and explain in the report the different steps taken to develop the requested software. This will be used by the marker to check inconsistencies or unclear points about the software. **The source code must be included as plain text (no images) in an appendix at the end of the report.**

Task 1.A

Using the most appropriate AI methods and a suitable programming language, implement a software application to solve the *Rare Disease and test problem* explained during the lectures. The program has as input the following probabilities:

P(d) : prior probability of having a disease
P(t | d) : probability that the test is positive given the person has the disease
P(¬t | ¬d) : probability that the test is negative given the person does not have the disease

where the meaning of the variables are:

d: the person has the disease
t: the test is positive

After the values for the previous probabilities are set the program should calculate the rest of necessary probabilities and calculate the probability of having the disease given the test was positive:

P(d | t)

You should be able to easily change the initial probability values and run the program to get the new result.

Task 1.B

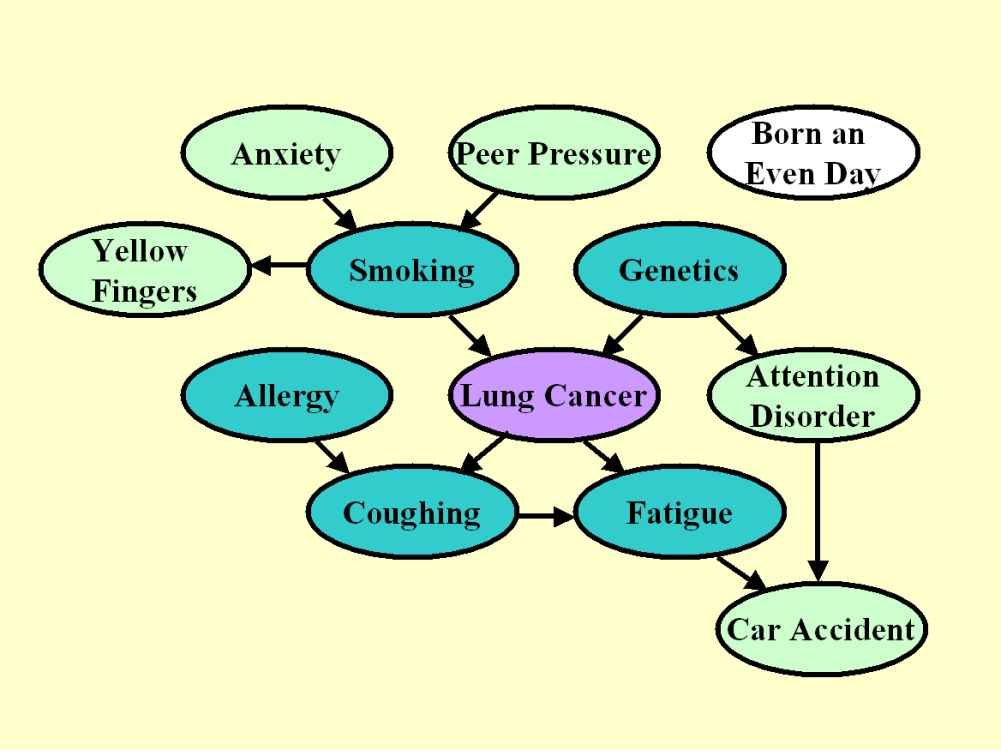
Using the most appropriate AI methods and a suitable programming language, implement a software application to solve a Medical Diagnosis problem similar to the *Burglary problem* explained during the lectures. The program should have as input the dataset shown in the link below, and the Bayes Network structure shown in the figure below – which uses the following random variables:

S: Smoking
YF: Yellow_Fingers
A: Anxiety
PP: Peer_Pressure
G: Genetics
AD: Attention_Disorder
BED: Born_an_Even_Day
CA: Car_Accident
F: Fatigue
A: Allergy
C: Coughing

Your program should first learn its **parameters** from data and then use **inference** to calculate the probability distribution of Smoking given Coughing and Fatigue:

P(S | C=True, F=True)

Please indicate in your report the methods used for implementing parameter learning and probabilistic inference.



Data=LUCAS0 Train: http://www.causality.inf.ethz.ch/data/LUCAS.html

2. TASK ON MARKOV MODELS

For the following task, select and critically appraise the most relevant technique(s) to solve the problem, justify your choice and explain in the report the different steps taken to develop the requested software. This will be used by the marker to check inconsistencies or unclear points about the software. **The source code must be included as plain text (no images) in an appendix at the end of the report.**

Task

Consider a heater with two possible unknown states, **ON** and **OFF**, placed in a room where the measurable temperature can be **Hot**, **Warm**, or **Cold**.
The probability between two consecutive time instants of remaining in the same state is **0.7** (e.g. if ON at time *t*, there are 70% chances of being ON again at time *t*+1), while the probability of switching to a different state is **0.3**.
In state **ON**, the probability of measuring a **Warm** or **Hot** temperature is **0.4**, while the probability of measuring **Cold** is **0.2**.
In state **OFF**, the probability of measuring a **Warm** or **Cold** temperature is **0.45**, while the probability of measuring **Hot** is **0.1**.

a) Implement and explain in the report a software application that, given in input any sequence of the above temperatures (i.e. the sequence can have any length), returns the probability of observing such sequence. Without any prior information, you can assume the initial states of the system to be equiprobable.

b) In the report, explain how you formalized the problem, with the aid of diagrams and tables, and describe step-by-step the operations and the numerical results of your software to compute the probability of observing the sequence **Cold-Warm-Hot-Warm-Cold**.