NATIONAL UNIVERSITY OF SINGAPORE

SCHOOL OF COMPUTING

FINAL ASSESSMENT FOR
Semester 2 AY2018/19

CS3243: INTRODUCTION TO ARTIFICIAL INTELLIGENCE

May 6, 2019 — Time Allowed: 2 Hours

INSTRUCTIONS TO CANDIDATES

1. This assessment paper contains **FIVE (5)** parts and comprises **20** printed pages, including this page.
2. Please fill in your **Student Number** below; **DO NOT WRITE YOUR NAME**.
3. Answer **ALL** questions as indicated. Unless explicitly said otherwise, you **must explain your answer**. Unexplained answers will be given a mark of 0.
4. Use the space provided to write down your solutions. If you need additional space to write your answers, we will provide you with draft paper. Clearly write down your **student number** (but not your name) and **question number** on the draft paper, and attach them to your assessment paper. Make sure you indicate on the **body of your paper as well** if you used draft paper to answer any question.
5. This is a **CLOSED BOOK** assessment.
6. You are allowed to use **NUS APPROVED CALCULATORS**.
7. For your convenience, we include an overview section of important definitions and algorithms from class at the end of this paper.

STUDENT NUMBER: ______________________

EXAMINER'S USE ONLY		
Part	Mark	Score
I	10	
II	10	
III	10	
IV	10	
V	10	
TOTAL	50	

In Parts I, II, III, IV and V you will find a series of short essay questions. For each short essay question, give your answer in the reserved space in the script.

Part I
Uninformed and Informed Search

(10 points) Short essay questions. Answer in the space provided on the script.

Consider the following setting. We are given a finite set of points $V = \{\vec{x}_1, \dots, \vec{x}_m\}$ in $\mathbb{R}^n$, two of which are goal nodes $G = \{\vec{q}, \vec{r}\}$ (we assume that $\vec{q} \neq \vec{r}$). The start node is some point $\vec{s} \in V$. We measure distance between the nodes using the Euclidean norm over $\mathbb{R}^n$: $\|\vec{x}\| = \sqrt{\sum_{i=1}^n x_i^2}$. Recall that $\|\cdot\|$ satisfies the triangle inequality: for all $\vec{x}, \vec{y} \in \mathbb{R}^n$, $\|\vec{x} + \vec{y}\| \leq \|\vec{x}\| + \|\vec{y}\|$.
There is a set of directed edges between points in V, denoted $E \subseteq V \times V$, and we set

$$c(\vec{x}_i, \vec{x}_j) = \begin{cases} \|\vec{x}_i - \vec{x}_j\| & \text{if } (\vec{x}_i, \vec{x}_j) \in E \\ \infty & \text{otherwise.} \end{cases}$$

1. (3 points) Given a point $\vec{x}_i \in V$, is the following heuristic admissible?

$$h_1(\vec{x}) = \max\{\|\vec{x} - \vec{q}\|, \|\vec{x} - \vec{r}\|\}$$

Prove this or provide a counterexample.

Solution: This is incorrect. Consider the case where $s = 0, x_2 = -0.5$, $\vec{q} = 1, \vec{r} = -1$. We have $h_1(x_2) = 1.5 > |x_2 - r| = 0.5$.

2. (3 points) Prove that the following heuristic is consistent.

$$h_2(\vec{x}) = \min\{\|\vec{x} - \vec{q}\|, \|\vec{x} - \vec{r}\|\}$$

You must provide a complete proof (not using any theorems shown in class/tutorials), using only the definition of consistency and properties of the Euclidean norm.

Solution: Given two points $\vec{x}_i, \vec{x}_j \in V$, let $D(\vec{x}_i, \vec{x}_j)$ be the shortest distance between $\vec{x}_i$ and $\vec{x}_j$. We need to show that for any two points $\vec{x}_i, \vec{x}_j$,

$$h_2(\vec{x}_i) \leq D(\vec{x}_i, \vec{x}_j) + h(\vec{x}_j)$$

If there is no path from $\vec{x}_i$ to $\vec{x}_j$ then we are done: $D(\vec{x}_i, \vec{x}_j) = \infty$. Let us next show the claim via induction on the length of the shortest path between $\vec{x}_i$ and $\vec{x}_j$. If it is of length 1 we are done: $D(\vec{x}_i, \vec{x}_j) = \|\vec{x}_i - \vec{x}_j\|$; next, suppose without loss of generality that $h_2(\vec{x}_j) = \|\vec{x}_j - \vec{q}\|$; then

$$h_2(\vec{x}_i) \leq \|\vec{x}_i - \vec{q}\| \leq \|\vec{x}_i - \vec{x}_j\| + \|\vec{x}_j - \vec{q}\| = D(\vec{x}_i, \vec{x}_j) + h_2(\vec{x}_j).$$

If there is a path of length k, let $\vec{x}_k$ be the last vector on the path from $\vec{x}_i$ to $\vec{x}_j$ before reaching $\vec{x}_j$; again, assume with no loss of generality that $h_2(\vec{x}_j) = \|\vec{x}_j - \vec{q}\|$; then

$$\begin{aligned} h_2(\vec{x}_i) \leq & D(\vec{x}_i, \vec{x}_k) + h_2(\vec{x}_k) \leq D(\vec{x}_i, \vec{x}_k) + \|\vec{x}_k - \vec{q}\| \leq D(\vec{x}_i, \vec{x}_k) + \|\vec{x}_k - \vec{x}_j\| + \|\vec{x}_j - \vec{q}\| \\ = & D(\vec{x}_i, \vec{x}_j) + h_2(\vec{x}_j) \end{aligned}$$

3. (4 points) Consider the following heuristic:

$$h_3(\vec{x}) = \frac{1}{B} \left| \sum_{i=1}^n (x_i - q_i) \times (x_i - r_i) \right|$$

Where $B = \max_{\vec{x},\vec{y} \in V} \|\vec{x} - \vec{y}\|$. Prove that h_3 is admissible. You must provide a complete proof (not using any theorems shown in class/tutorials), using only the definition of admissibility and properties of the Euclidean norm.

Hint: You may consider using the **Cauchy-Schwartz inequality**: for any two vectors $\vec{a}, \vec{b} \in \mathbb{R}^n$

$$\left(\sum_{i=1}^{n} a_i b_i\right)^2 \le \left(\sum_{i=1}^{n} a_i^2\right) \times \left(\sum_{i=1}^{n} b_i^2\right)$$

Solution:

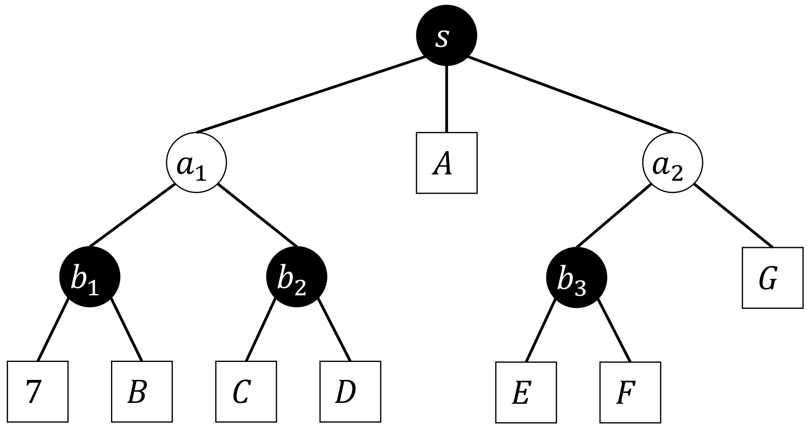
$$\begin{aligned}
h_3(\vec{x}) =& \frac{1}{B}\left|\sum_{i=1}^{n}(x_i - q_i) \times (x_i - r_i)\right| \\
=& \frac{1}{B}\sqrt{\left(\sum_{i=1}^{n}(x_i - q_i) \times (x_i - r_i)\right)^2} \\
\le& \frac{1}{B}\sqrt{\left(\sum_{i=1}^{n}(x_i - q_i)^2\right) \times \left(\sum_{i=1}^{n}(x_i - r_i)^2\right)} \\
=& \frac{1}{B}\sqrt{\sum_{i=1}^{n}(x_i - q_i)^2} \times \sqrt{\sum_{i=1}^{n}(x_i - r_i)^2} \\
=& \frac{1}{B}\|\vec{x} - \vec{q}\| \times \|\vec{x} - \vec{r}\| \qquad (1)
\end{aligned}$$

Finally, assume with no loss of generality that $\|\vec{x} - \vec{r}\| \le \|\vec{x} - \vec{q}\|$; (1) is upper bounded by $\|\vec{x} - \vec{r}\| = h_2(\vec{x})$, which concludes the proof.

Part II
Adversarial Search

(10 points) Short essay questions. Answer in the space provided on the script.

1. (7 points) Consider the minimax search tree shown below:



The MAX player controls the black nodes (s, b_1, b_2 and b_3) and the MIN player controls the white nodes a_1 and a_2. Square nodes are terminal nodes with utilities specified with respect to the MAX player; the values A, B, C, D, E, F, G are all non-negative integers. Suppose that we use the α-β pruning algorithm (reproduced in Figure VI.7), in the direction from **right to left** to prune the search tree. For each of the following cases, mark the box in the "True" column if the condition on the left is true, and the "False" column otherwise. You do not need to explain your answers.

Solution:

	True	False
If $B \geq \max\{C, D\}$ then some arc must be pruned	✓	☐
If $G \geq C$ then some arc must be pruned	☐	✓
If $A \geq \max\{C, D\}$ then some arc must be pruned	✓	☐
If $F \geq G$ then some arc must be pruned	✓	☐
If $7 \geq \min\{C, D\}$ then some arc must be pruned	☐	✓
If $B \leq 7$ then some arc must be pruned	☐	✓
If $G \geq \max\{C, D\}$ then some arc must be pruned	☐	✓

Let us set

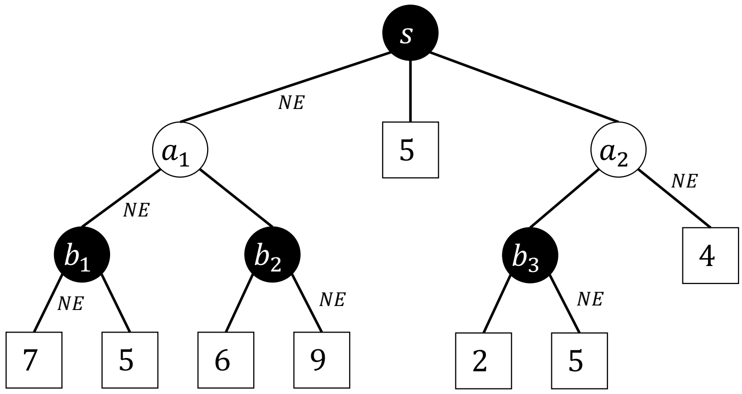
$$A = 2, B = 4, C = 1, D = 5, E = F = 4, G = 100.$$

No arc gets pruned for these values, and the inequalities for the false claims all hold. The first claim is true. If $B \geq \max\{C, D\}$ then the MIN player wil prune the arc $(b_1, 7)$. The third claim is true. Note that $\alpha(s) \geq A$ after scanning through the options of a_2 and A, thus if $A \geq \max\{C, D\}$, which is the value of $T(b_2)$, the arc (a_1, b_1) gets pruned. If $F \geq G$ then the arc from (b_3, E) gets pruned.

2. (3 points) Consider the minimax search tree in the solution box below. The MAX player controls the black nodes (s, b_1, b_2 and b_3) and the MIN player controls the white nodes a_1 and a_2. Square nodes are terminal nodes with utilities specified with respect to the MAX player.

   Mark down each arc used in a subgame-perfect Nash equilibirum (by writing `NE` next to it), and write down the utility of the MAX player under this equilibrium. You do not need to explain your answer to this question.

Solution:



2 points for marking down correct arcs. -0.5 point for each incorrectly marked arc (either false positive or false negative). 1 point for writing down correct game value (7 in this case).

Part III
Constraint Satisfaction Problems

(10 points) Short essay questions. Answer in the space provided on the script.
Figure III.1 represents a map of a country with six geographical regions (A, B, C, D, E, F). We must color each state red (R), green (G) or blue (B). Adjacent regions **cannot be of the same color**. To solve this problem, we use the backtracking search algorithm (Figure VI.8).

- Whenever a value is assigned, we only use the forward-checking operation (in the INFERENCE stage). No other constraint-propagation operation (such as AC3) is performed.
- When selecting an unassigned variable (the SELECT-UNASSIGNED-VARIABLE stage) the algorithm uses the most constrained-variable, and then the most-constraining-variable heuristic to break ties. If there are still ties, the algorithm breaks ties in alphabetical order.
- When selecting values (the ORDER-DOMAIN-VALUES stage), it uses the least-constraining-value heuristic. Whenever several values are tying for selection, the algorithm selects them in the following order: red, green, blue.

1. (2 points) Which variable will be selected first by the algorithm? What value will be assigned to it? Why?

> **Solution:** The algorithm will assign a value to D: all variables are equally constrained (all colors allowed), but D is the most constraining. The color R is assigned to D.

2. (2 points) Which values of which variable domains does the forward-checking operation then remove?

> **Solution:** In the next step, all variable domains for A, B, C, E, F will have the value R removed.

3. (2 points) Which variable will be selected next? Why? Which value will be assigned to this variable? Why?

> **Solution:** Next, all variable domains for A, B, C, E, F are the same, thus we see who's the most constraining. In this case, it is B and C; breaking ties alphabetically in favor of B, we then assign a value for B. We assign a value of G to it.

4. (2 points) Which values of which variable domains does the forward-checking operation then remove?

> **Solution:** The domains of A and C are reduced to B.

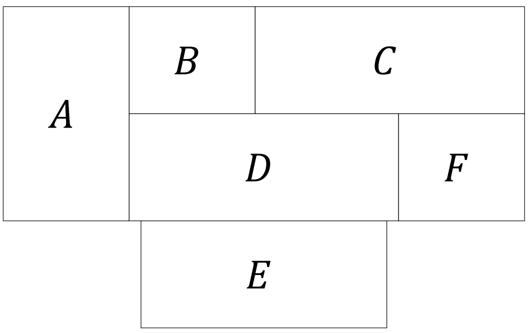
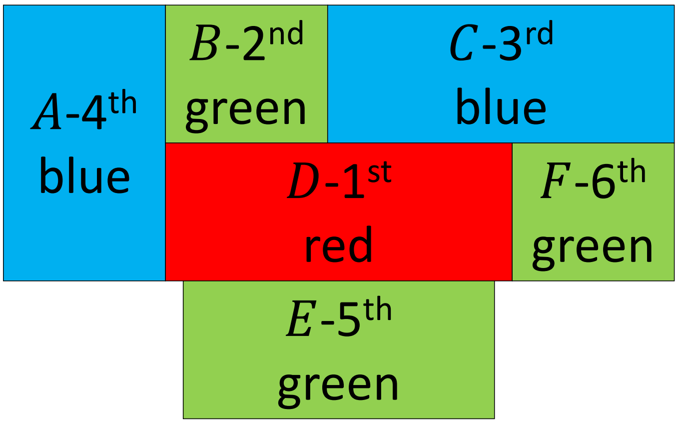


Figure III.1: Map diagram for graph coloring problem.

5. (2 points) Show the coloring outputted by the algorithm in the figure below (write 'red', 'green' or 'blue' in each box to represent the coloring).

Solution:



Part IV
Logical Agents

(10 points) Short essay questions. Answer in the space provided on the script.
NUS offers financial aid to students based on the following criteria. In order to be eligible for financial aid from NUS you must

(a) be enrolled to a bachelor's degree programme at NUS;

(b) earn no more than S$2,700/mth if you are a Singaporean/PR, and no more than S$1,200 otherwise;

(c) be making satisfactory progress in your studies.

Students whose GPA is at least 3.0, or who have a reference letter from a professor are considered to be making satisfactory progress. Students who work as part-time programmers make less than S$1,200/mth, and less than S$2,700/mth.
Alice is currently enrolled to a bachelor's degree programme; she is not a Singaporean/PR; she works as a part-time programmer, and has a reference letter from her professor.

1. (4 points) Represent the above statements (those describing the financial aid eligibility rules, and those describing Alice's properties) as a first-order logic knowledge base.

Solution: Answer should be logically equivalent (up to function names) to:

1. $\forall x : \texttt{Bach}(x) \wedge ((\texttt{SPR}(x) \wedge \texttt{Less2.7k}(x)) \vee (\neg\texttt{SPR}(x) \wedge \texttt{Less1.2k}(x))) \wedge \texttt{Progress}(x) \Rightarrow \texttt{Elig}(x)$
2. $\forall x : \texttt{GPA}(x) \Rightarrow \texttt{Progress}(x)$
3. $\forall x : \texttt{Ref}(x) \Rightarrow \texttt{Progress}(x)$
4. $\forall x : \texttt{PTCoder}(x) \Rightarrow \texttt{Less2.7k}(x)$
5. $\forall x : \texttt{PTCoder}(x) \Rightarrow \texttt{Less1.2k}(x)$.
6. $\texttt{Bach}(Alice)$
7. $\neg\texttt{SPR}(Alice)$
8. $\texttt{Ref}(Alice)$
9. $\texttt{PTCoder}(Alice)$

2. (3 points) Convert the above knowledge base to a Skolemized CNF form.

Solution: Answer should be logically equivalent (up to function names) to:

1. $\neg\texttt{Bach}(w) \vee \neg\texttt{SPR}(w) \vee \neg\texttt{Less2.7k}(w) \vee \neg\texttt{Progress}(w) \vee \texttt{Elig}(w)$
2. $\neg\texttt{Bach}(x) \vee \texttt{SPR}(x) \vee \neg\texttt{Less1.2k}(x) \vee \neg\texttt{Progress}(x) \vee \texttt{Elig}(x)$
3. $\neg\texttt{GPA}(y) \vee \texttt{Progress}(y)$
4. $\neg\texttt{Ref}(z) \vee \texttt{Progress}(z)$
5. $\neg\texttt{PTCoder}(q) \vee \texttt{Less2.7k}(q)$
6. $\neg\texttt{PTCoder}(r) \vee \texttt{Less1.2k}(r)$.
7. $\texttt{Bach}(Alice)$
8. $\neg\texttt{SPR}(Alice)$
9. $\texttt{Ref}(Alice)$
10. $\texttt{PTCoder}(Alice)$

3. (3 points) Use FOL Resolution to infer that Alice is eligible for financial aid. In each line in the table below, write the two FOL sentences you wish to resolve as (S_1) and (S_2); **you do not need to write the complete sentence, but rather refer to the sentence number used in the previous question**. The resolvent should appear in the marked column; **you can enumerate the resolvents and refer to them in later resolutions for your convenience**. The required substitution should be written in the last column. Note that you might not need to fill all table rows.

Solution:

Query we need to infer (phrased in FOL logic) $\alpha = \texttt{Elig}(Alice)$

S_1	S_2	Resolvent	Substitution
(2)	(7)	$\texttt{SPR}(Alice) \vee \neg\texttt{Less1.2k}(Alice) \vee \neg\texttt{Progress}(Alice) \vee \texttt{Elig}(Alice)$ (11)	$x \leftarrow Alice$
(11)	(8)	$\neg\texttt{Less1.2k}(Alice) \vee \neg\texttt{Progress}(Alice) \vee \texttt{Elig}(Alice)$ (12)	none
(4)	(9)	$\texttt{Progress}(Alice)$ (13)	$z \leftarrow Alice$
(13)	(12)	$\neg\texttt{Less1.2k}(Alice) \vee \texttt{Elig}(Alice)$ (14)	none
(10)	(6)	$\texttt{Less1.2k}(Alice)$ (15)	$r \leftarrow Alice$
(15)	(14)	$\texttt{Elig}(Alice)$ (16)	none
(16)	$\neg\alpha$	[]	none

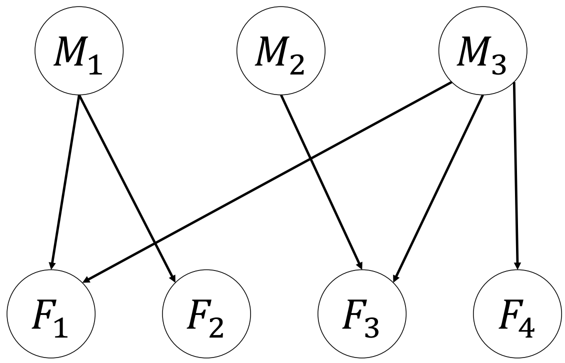
Part V
Uncertainty and Bayesian Networks

(10 points) Short essay questions. Answer in the space provided on the script.
An IT technician receives a student's malfunctioning laptop for inspection; the technician suspects three types of malware M_1, M_2 and M_3 to be the cause of the malfunction. It is assumed that the probabilities of getting the

three malwares are completely independent of one another. There are four system files F_1, F_2, F_3, F_4 that may be affected by the malware infection. F_1 may only be altered by M_1 and M_3; F_2 can only be altered by M_1; F_3 may only be altered by M_2 and M_3; F_4 may only be altered by M_3. Assume all random variables are Boolean: they are either 'true' or 'false'. System files are not guaranteed to be changed by the malware; this depends on the (unknown, presumed to be random) system configuration.

1. (3 points) Draw a Bayesian network with a minimal number of arcs for the problem described above.

Solution:



2. (3 points) Suppose that the technician is certain that the computer is not infected by M_3 (i.e. $M_3 = 0$). Which of the statements below is true? Check the appropriate boxes below.

Solution:

Given that $M_3 = 0$:

	True	False
F_1 and F_4 are independent	✓	☐
F_1 and F_2 are independent	☐	✓
F_1 and F_3 are independent	✓	☐

3. (4 points) Suppose that:

$$\Pr[M_1 = 1] = 0.2; \Pr[M_2 = 1] = 0.3, \Pr[M_3 = 1] = 0.4$$

and the following conditional probabilities are known:

$\Pr[F_1 \mid \neg M_1 \wedge \neg M_3] = 0.01$ $\Pr[F_1 \mid M_1 \wedge \neg M_3] = 0.2$ $\Pr[F_1 \mid \neg M_1 \wedge M_3] = 0.3$ $\Pr[F_1 \mid M_1 \wedge M_3] = 0.6$	$\Pr[F_2 \mid \neg M_1] = 0.02$ $\Pr[F_2 \mid M_1] = 0.9$	$\Pr[F_3 \mid \neg M_2 \wedge \neg M_3] = 0.1$ $\Pr[F_3 \mid M_2 \wedge \neg M_3] = 0.5$ $\Pr[F_3 \mid \neg M_2 \wedge M_3] = 0.1$ $\Pr[F_3 \mid M_2 \wedge M_3] = 0.5$	$\Pr[F_4 \mid \neg M_3] = 0.2$ $\Pr[F_4 \mid M_3] = 0.3$

Suppose that we know that F_1 and F_4 were corrupted, but F_2 and F_3 were not; furthermore, we know that M_2 is not infecting the laptop. Which is likelier, that $M_1 = 1$ or that $M_3 = 1$? You must show your work.

Solution: We need to compute $\Pr[M_1 \mid F_1, F_2, F_3, F_4]$ and $\Pr[M_3 \mid F_1, F_2, F_3, F_4]$.

$$\begin{aligned}
&\Pr[M_1 \mid F_1, \neg F_2, \neg F_3, F_4, \neg M_2] = \\
&\frac{\Pr[F_1, \neg F_2, \neg F_3, F_4, \neg M_2 \mid M_1] \times \Pr[M_1]}{\Pr[F_1, \neg F_2, \neg F_3, F_4, \neg M_2]} = \\
&\alpha \Pr[F_1, \neg F_2, \neg F_3, F_4, \neg M_2 \mid M_1] \times \Pr[M_1] = \\
&\alpha \Pr[F_1, \neg F_2, \neg F_3, F_4 \mid M_1, \neg M_2] \times \Pr[M_1] \times \Pr[\neg M_2] = \\
&\alpha \Pr[F_1, \neg F_2, \neg F_3, F_4 \mid M_1, \neg M_2, M_3] \times \Pr[M_3] \times \Pr[M_1] \times \Pr[\neg M_2] + \\
&\alpha \Pr[F_1, \neg F_2, \neg F_3, F_4 \mid M_1, \neg M_2, \neg M_3] \times \Pr[\neg M_3] \times \Pr[M_1] \times \Pr[\neg M_2] = \\
&\alpha \Pr[F_1 \mid M_1, M_3] \times \Pr[\neg F_2 \mid M_1] \times \Pr[\neg F_3 \mid \neg M_2, M_3] \times \Pr[F_4 \mid M_3] \times \Pr[M_3] \times \Pr[M_1] \times \Pr[\neg M_2] + \\
&\alpha \Pr[F_1 \mid M_1, \neg M_3] \times \Pr[\neg F_2 \mid M_1] \times \Pr[\neg F_3 \mid M_2, \neg M_3] \times \Pr[F_4 \mid \neg M_3] \times \Pr[\neg M_3] \times \Pr[M_1] \times \Pr[\neg M_2] = \\
&\alpha 0.6 \times 0.1 \times 0.9 \times 0.3 \times 0.4 \times 0.2 \times 0.7 + \\
&\alpha 0.2 \times 0.1 \times 0.9 \times 0.2 \times 0.6 \times 0.2 \times 0.7 \simeq \\
&0.0012\alpha
\end{aligned}$$

We repeat the exercise for M_3:

$$\begin{aligned}
&\Pr[M_3 \mid F_1, \neg F_2, \neg F_3, F_4, \neg M_2] = \\
&\frac{\Pr[F_1, \neg F_2, \neg F_3, F_4, \neg M_2 \mid M_3] \times \Pr[M_3]}{\Pr[F_1, \neg F_2, \neg F_3, F_4, \neg M_2]} = \\
&\alpha \Pr[F_1, \neg F_2, \neg F_3, F_4, \neg M_2 \mid M_3] \times \Pr[M_3] = \\
&\alpha \Pr[F_1, \neg F_2, \neg F_3, F_4 \mid M_3, \neg M_2] \times \Pr[M_3] \times \Pr[\neg M_2] = \\
&\alpha \Pr[F_1, \neg F_2, \neg F_3, F_4 \mid M_1, \neg M_2, M_3] \times \Pr[M_3] \times \Pr[M_1] \times \Pr[\neg M_2] + \\
&\alpha \Pr[F_1, \neg F_2, \neg F_3, F_4 \mid \neg M_1, \neg M_2, M_3] \times \Pr[M_3] \times \Pr[\neg M_1] \times \Pr[\neg M_2] = \\
&\alpha \Pr[F_1 \mid M_1, M_3] \times \Pr[\neg F_2 \mid M_1] \times \Pr[\neg F_3 \mid \neg M_2, M_3] \times \Pr[F_4 \mid M_3] \times \Pr[M_3] \times \Pr[M_1] \times \Pr[\neg M_2] + \\
&\alpha \Pr[F_1 \mid \neg M_1, M_3] \times \Pr[\neg F_2 \mid \neg M_1] \times \Pr[\neg F_3 \mid M_2, M_3] \times \Pr[F_4 \mid M_3] \times \Pr[M_3] \times \Pr[\neg M_1] \times \Pr[\neg M_2] = \\
&\alpha 0.6 \times 0.1 \times 0.9 \times 0.3 \times 0.4 \times 0.2 \times 0.7 + \\
&\alpha 0.3 \times 0.98 \times 0.5 \times 0.3 \times 0.4 \times 0.8 \times 0.7 \simeq \\
&0.0107\alpha
\end{aligned}$$

Thus it is far likelier that $M_3 = 1$ than that $M_1 = 1$.