

points:	grade:	exam viewed on:
---------	--------	-----------------

Exam PROBE-KLAUSUR
Introduction to Artificial Intelligence, SS 19, U Stuttgart

Prof. Dr. Marc Toussaint

September 13, 2019

Name:
Matrikelnummer:

**DO NOT OPEN THE EXAM BEFORE THE
ANNOUNCEMENT!**

DIE KLAUSUR NICHT VOR DER ANSAGE ÖFFNEN!

- You have 90 minutes. The total number of points is printed on the last page.
- Write your name on all sheets.
- Put your student ID (*Studentenausweis*) next to you, so we can check it during the exam.
- **You may only use your pen and the given paper.** Don't use pencil or red colour. No other materials or devices (no textbook, script, calculator, or mobile) are allowed.
- In case you need extra paper, please raise your hand and you will be provided extra sheets. All extra sheets need to be handed in together with the exam.
- **Try to write all your answers in the boxes!** By default we will not grade or correct anything outside the boxes. (If you indicate it explicitly, we will also check the rest.) The free space and backside of pages is for your use as scrap paper.

-
- Sie haben 90 Minuten.
 - Schreiben Sie Ihren Namen auf jedes Blatt der Klausur.
 - Legen Sie Ihren Studentenausweis neben sich, damit wir ihn während der Klausur einsehen können.
 - **Sie dürfen ausschließlich Kugelschreiber/Tintenschreiber und das ausgehändigte Papier nutzen.** Andere Hilfsmittel, wie Bücher, Skript oder Mobiltelefone/elektronische Geräte sind nicht erlaubt.
 - Falls Sie zusätzliches Papier benötigen melden Sie sich per Hand. Jedes ausgehändigte Papier muss mit der Klausur abgegeben werden.
 - **Versuchen Sie alle Antworten in die vorgesehenen Boxen zu schreiben!** Alles außerhalb der Boxen wird, im Normalfall, nicht korrigiert oder beachtet. (Nur wenn Sie es explizit anmerken schauen wir auch dort.) Der freie Raum und die Rückseite der Seiten dient Ihnen als Schmierpapier.

Question 1 — Constraint propagation and CSPs – PUBLIC (6Pts)

Consider three integer variables $X, Y, Z \in \{1, 2, 3, 4, 5\}$ and three constraints

$$X < Y, \quad X + Y + Z < 7, \quad Y = 2Z$$

- (i) Provide the result of constraint propagation (until convergence) by providing the remaining domain sets for each variable:

X:

Y:

Z:

- (ii) After constraint propagation, backtracking search will do a variable assignment to search for a feasible solution. In the lecture we learnt about two heuristics for choosing the next variable to be assigned. In our case, which variable will be chosen to be assigned a value according to these heuristics? [X, Y, or Z]

- (iii) How many solutions do exist for this CSP?

- (i) Schreiben Sie das Bayes-Theorem auf.
- (ii) Gegeben drei Zufallsvariablen X, Y, Z und deren Verbundwahrscheinlichkeit $P(X, Y, Z)$. Was ist die Definition von $P(X|Y)$ gegeben $P(X, Y, Z)$?
- (iii) In A^* , sei $c(n, n')$ die wahren Kosten eines einzelnen Schrittes von n nach n' . Was ist die Definition einer *konsistenten* Heuristik? (Jede konsistente Heuristik ist auch zulässig.) Schreiben Sie die Bedingung, die $h(n)$ erfüllen muss, in die Box.

Wir haben drei Integer-Variablen $X, Y, Z \in \{1, 2, 3, 4, 5\}$ und drei Bedingungen $X < Y, X + Y + Z < 7, Y = 2Z$. (i) Geben Sie das Resultat von *Constraint Propagation* an, indem Sie für jede Variable die verbleibende *Domain* angeben. (ii) Gemäß den Heuristiken, die in der Vorlesung eingeführt wurden, welche Variable wird als nächstes von *backtracking* ausgewählt, um einen Wert zuzuweisen? (iii) Wieviele Lösungen existieren für dieses CSP?

Question 2 — Tree search – PUBLIC (6Pts)

Assuming a tree search with

- maximum branching factor b of the search tree,
- depth d of the least-cost solution,
- maximum depth m of the state space.

What is the space complexity of the following tree search methods (neglecting a +1 or -1 in the expressions is ok):

- breadth first search?
- depth first search?
- iterative deepening search?

Which of the three is optimal (i.e., computes an optimal solution assuming every step cost is 1). Answer [Y/N].

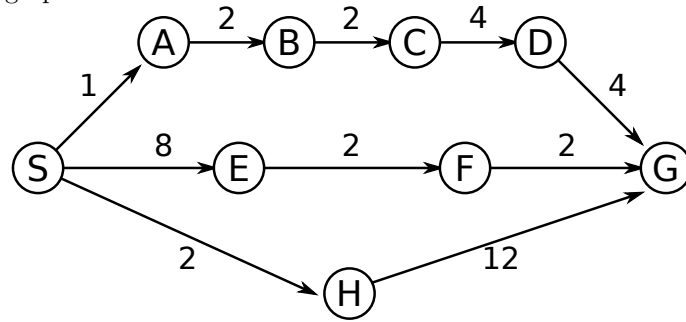
- breadth first search?
- depth first search?
- iterative deepening search?

Wir betrachten eine Baumsuche mit maximalem Verzweigungsgrad b , Tiefe d der optimalen Lösung, und maximaler Tiefe m des Zustandsraums. Was ist die Speicherkomplexität von a) Breitensuche, b) Tiefensuche, c) iterativer Tiefensuche.

Welche der drei Folgenden ist optimal (d.h., berechnet die optimale Lösung unter der Annahme, dass jeder Schritt die Kosten 1 hat.) d) Breitensuche, e) Tiefensuche, f) iterative Tiefensuche.

Question 3 — A* – PUBLIC (12Pts)

Consider the following state graph:



Edges are labeled with the cost $c(s, s')$ for a state transition; the start state is S ; the goal state is G . Assume a *constant* heuristic $\forall s \neq G : h(s) = 1$ with value 1 for all states except for the goal state, where $h(G) = 0$.

Provide below the *priority queue* in every iteration of A*. (More precisely: write the state of the queue *before* the goal test and expand operation.) In each box, use the notation “ $X:g+h$ ” for an entry of the queue, where X denotes the state, g the cost-so-far, and h the heuristic. Left entries should have higher priority (lower $g + h$).

Step 0:	S:0+1		
Step 1:			
Step 2:			
Step 3:			
Step 4:			
Step 5:			
Step 6:			
Step 7:			
Step 8:			

In step 7, an F -node is inserted into the priority queue *before* the previously inserted, non-optimal G -node. Provide equations and inequalities which prove $f(F) \leq f(G_{\text{non-opt}})$ for an admissible heuristic, where $G_{\text{non-opt}}$ is such a non-optimal G -node.

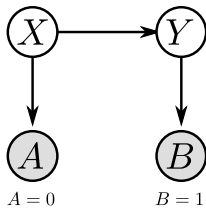
Im gegebenen Zustandsgraph sind an den Kanten die Kosten $c(s, s')$ für einen Übergang zwischen zwei Zuständen gegeben. S ist der Start-Zustand; G der Ziel-Zustand. Die Heuristik ist für alle Zustände konstant 1, außer für den Ziel-Zustand G , wo sie 0 ist: $\forall s \neq G : h(s) = 1, h(G) = 0$.

Schreiben Sie *priority queue* in jeder Iteration von A* nieder. Genauer: Schreiben Sie den Zustand der Queue *vor* der *goal test* und *expand*-Operation nieder. Verwenden Sie für einen Eintrag in der Queue die Notation “ $X:g+h$ ”, wobei X der Name des Zustands, g die cost-so-far, und h die Heuristik bezeichnet. Links sollten Einträge höherer Priorität stehen.

In Schritt 7 wird ein F -Knoten “*vor*” den bereits gefundenen nicht-optimalen G -Knoten in die *priority queue* eingegliedert. Schreiben Sie Gleichungen und Ungleichungen die beweisen, dass $f(F) \leq f(G_{\text{non-opt}})$; wobei $G_{\text{non-opt}}$ ein solcher nicht-optimaler G -Knoten ist und die Heuristik *admissible* ist.

Question 4 — Bayesian Inference – PUBLIC (6Pts)

Given the graphical model (actually a 2-step HMM)



$$P(X) = \begin{bmatrix} .5 \\ .5 \end{bmatrix}$$

$$P(A|X) = P(Y|X) = P(B|Y) = \begin{bmatrix} .8 & .2 \\ .2 & .8 \end{bmatrix}$$

where $A = 0$ and $B = 1$ are observed.(i) Compute $P(Y | A = 0, B = 1) \propto$

Note that this may ignore normalization factors. Provide the answer in the box, ideally in terms of an explicit matrix/vector equation, where you may use \circ to denote the element-wise vector product. Alternatively, provide an explicit table of two numbers proportional to $P(Y | A = 0, B = 1)$.

(ii) Compute $P(A = 0, B = 1) =$ Provide the single number for the observation likelihood $P(A = 0, B = 1)$ in the box.

TOTAL POINTS=30

Im gegebenen graphischen Modell (ein 2-Schritt-HMM) werden $A = 0$ und $B = 1$ beobachtet. (i) Berechnen Sie $P(Y | A = 0, B = 1)$ modulo Normalisierung. Idealerweise können Sie die Antwort als explizite Matrix/Vector-Gleichung angeben, wobei \circ ein element-weise Produkt bezeichnet. Alternativ geben Sie eine Tabelle mit zwei Zahlen proportional zu $P(Y | A = 0, B = 1)$ an. (ii) Geben Sie die Beobachtungs-Likelihood $P(A = 0, B = 1)$ als einzelne Zahl in der Box an.