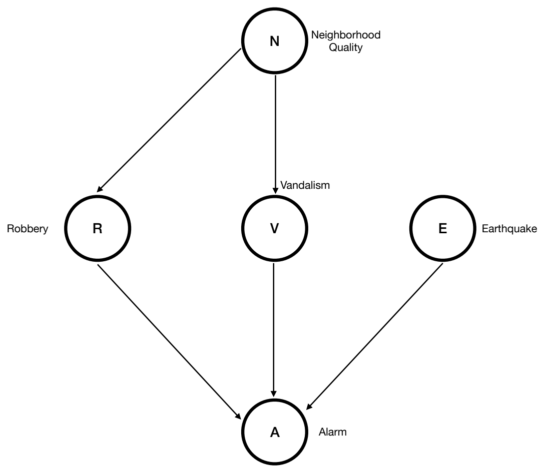
6. Use the Bayesian network in Figure 2 together with the conditional probability tables below to answer the following questions. If you do not have a hand-held calculator with you, make sure you set up the solution to the problems appropriately for partial credit.

(a) Write the formula for the full joint probability distribution $P(A, R, V, E, N)$ in terms of (conditional) probabilities using the chain rule. **[1p]**

(b) Write the formula for the full joint probability distribution $P(A, R, V, E, N)$ in terms of (conditional) probabilities derived from the bayesian network below. **[1p]**

(c) What is the probability, $P(a, r, \neg v, e, N = high)$? **[1p]**

(d) What is the probability, $P(a \mid \neg e, v)$? **[2p]**



A	R	V	E	$P(A \mid R, V, E)$
T	T	T	T	0.98
F	T	T	T	0.02
T	T	T	F	0.75
F	T	T	F	0.25
T	T	F	T	0.65
F	T	F	T	0.35
T	T	F	F	0.70
F	T	F	F	0.30
T	F	F	F	0.05
F	F	F	F	0.95
T	F	F	T	0.10
F	F	F	T	0.90
T	F	T	F	0.55
F	F	T	F	0.45
T	F	T	T	0.35
F	F	T	T	0.65

R	N	$P(R \mid N)$
T	high	.2
F	high	.8
T	low	.75
F	low	.25

V	N	$P(V \mid N)$
T	high	.2
F	high	.8
T	low	.75
F	low	.25

E	P(E)
T	.05
F	.95

N	P(N)
high	.75
low	.25

(a) Write the formula for the full joint probability distribution $P(A, R, V, E, N)$ in terms of (conditional) probabilities using the chain rule. **[1p]**

a.) $\mathbf{P}(A, R, V, E, N) =$

$$\mathbf{P}(A \mid R, V, E, N)\mathbf{P}(R \mid V, E, N)\mathbf{P}(V \mid E, N), \mathbf{P}(E \mid N)\mathbf{P}(N)$$

(b) Write the formula for the full joint probability distribution $P(A, R, V, E, N)$ in terms of (conditional) probabilities derived from the bayesian network below. **[1p]**

b.) Choose a topological order for nodes:
(N, R, V, E, A)

$$\mathbf{P}(N)\mathbf{P}(R \mid N)\mathbf{P}(V \mid N)\mathbf{P}(E)\mathbf{P}(A \mid R, V, E)$$

(c) What is the probability, $P(a, r, \neg v, e, N = high)$? **[1p]**

c.) $P(a, r, \neg v, e, N = high) =$

$$P(N = high)P(r \mid N = high)P(\neg v \mid N = high)P(e)P(a \mid r, \neg v, e)$$

$$= 0.75 * 0.2 * 0.8 * 0.05 * 0.65 = 0.0039$$

(d) What is the probability, $P(a \mid \neg e, v)$? **[2p]**

d). An Inference Example

$$\mathbf{P}(X \mid \mathbf{e}) = \alpha * \mathbf{P}(X, \mathbf{e}) = \alpha * \sum_{\mathbf{y}} \mathbf{P}(X, \mathbf{e}, \mathbf{y})$$

$X = \{Alarm\}$
$\mathbf{E} = \{Vandalism, Earthquake\}$
$\mathbf{e} = \{\neg earthquake, vandalism\}$
$\mathbf{Y} = \{Robbery, Nquality\}$

Query: $P(alarm \mid \neg earthquake, vandalism)$

$$P(alarm \mid \neg earthquake, vandalism) = \frac{P(alarm, \neg earthquake, vandalism)}{P(\neg earthquake, vandalism)}$$

$$= \alpha \, P(alarm, \neg earthquake, vandalism)$$

$$= \alpha \sum_{\mathbf{r},\mathbf{n}} P(alarm, \neg earthquake, vandalism, \mathbf{r}, \mathbf{n})$$

$$= \alpha[P(a, \neg e, v, r, N = h) + P(a, \neg e, v, r, N = l) + P(a, \neg e, v, \neg r, N = h) + P(a, \neg e, v, \neg r, N = l)]$$

where $\alpha = \dfrac{1}{P(\neg e, v)} = \dfrac{1}{\sum_{\mathbf{a},\mathbf{r},\mathbf{n}} P(\neg e, v, \mathbf{a}, \mathbf{r}, \mathbf{n})}$

LINKÖPING UNIVERSITY

3

$$= \alpha[P(a, \neg e, v, r, N = h) + P(a, \neg e, v, r, N = l) + P(a, \neg e, v, \neg r, N = h) + P(a, \neg e, v, \neg r, N = l)]$$

where $\alpha = \dfrac{1}{P(\neg e, v)} = \dfrac{1}{\sum_{\mathbf{a},\mathbf{r},\mathbf{n}} P(\neg e, v, \mathbf{a}, \mathbf{r}, \mathbf{n})}$

We know how to compute each probability:

$$P(a, \neg e, v, r, N = h)$$

$$= P(N = h)P(r \mid N = high)P(v \mid N = high)P(\neg e)P(a \mid r, v, \neg e)$$

$$= 0.75 * 0.2 * 0.2 * 0.95 * 0.75 = 0.021375$$

And

$$\sum_{\mathbf{a},\mathbf{r},\mathbf{n}} P(\neg e, v, \mathbf{a}, \mathbf{r}, \mathbf{n}) = P(\neg e, v, a, r, N = h) + \ldots + P(\neg e, v, \neg a, \neg r, N = l)$$

To avoid computing α in this way, compute $\mathbf{P}(A \mid \neg e, v) = \alpha \mathbf{P}(A \neg e, v)$ instead where,

$$\alpha = \frac{1}{P(a, \neg e, v) + P(\neg a, \neg e, v)}$$

Question:DPLL [Total points: **[6p]**]

The following questions pertain to the extended Davis-Putnam algorithm (DPLL) considered in the course book. DPLL takes as input a formula in propositional logic, transforms it to its equivalent conjunctive normal form (CNF) representation and determines whether the formula is satisfiable or not. The questions pertain to the different heuristics used in DPLL.

(When asked to "complete the table . . . " in the questions below, write your own complete table in the answer page. Do not fill in these tables on the exam page.)

a. Complete the table below by applying the Splitting rule to variable B and CNF: [**1p**]

$$\alpha = (A \lor B \lor C \lor D) \land (\neg A \lor B \lor \neg C) \land (\neg B \lor \neg C \lor D) \land (A \lor \neg B \lor C \lor D)$$
$$\land (B \lor \neg C \lor \neg D) \land (\neg A \lor \neg C \lor D) \land (\neg A \lor \neg D) \land (A \lor \neg B)$$

Table 1:

Heuristic	Model_1	Simplified CNF_1	Model_2	Simplified CNF_2
Initial call	$\{\}$	α	-	-
$SR[B]$	$\{B : True\}$		$\{B : False\}$	

b. Complete the table below by applying the Unit Clause Heuristic (UCH) as long as possible to the CNF: [**1.5p**]

$$\alpha = \neg P \land (\neg Q \lor R \lor S) \land (\neg P \lor Q) \land (P \lor \neg R)$$

Use one row for each call and state the variable the heuristic is being applied to in the 1st column, the extension to the model in the 2nd column and the resulting formula in the 3rd column. Add as many additional rows as required.

Table 2:

Heuristic	Model	Simplified CNF
Initial call	$\{\}$	α
$UCH[-]$		
$UCH[-]$		
$UCH[-]$		

c. Complete the table below by applying the Pure Symbol Heuristic (PSH) as long as possible to the CNF: [**1.5p**]

$$\alpha = (\neg P \lor Q) \land (R \lor \neg Q) \land (M \lor \neg N) \land (N \lor \neg M) \land (Q \lor N)$$

Use one row for each call and state the variable the heuristic is being applied to in the 1st column, the extension to the model in the 2nd column and the resulting formula in the 3rd column. Add as many additional rows as required.

Table 3:

Heuristic	Model	Simplified CNF
Initial call	$\{\}$	α
$PSH[-]$		
$PSH[-]$		
$PSH[-]$		

d. Suppose we had three formulas in propositional logic, $F1, F2$ and $F3$. Explain how one would show whether $F1 \land F2 \models F3$ using the DPLL algorithm. In doing this, it is important to show the formal relation between $\models$ and satisfaction. [**2p**]

a. Complete the table below by applying the Splitting rule to variable B and CNF: [**1p**]

$$\alpha = (A \lor B \lor C \lor D) \land (\neg A \lor B \lor \neg C) \land (\neg B \lor \neg C \lor D) \land (A \lor \neg B \lor C \lor D)$$
$$\land (B \lor \neg C \lor \neg D) \land (\neg A \lor \neg C \lor D) \land (\neg A \lor \neg D) \land (A \lor \neg B)$$

Table 1:

Heuristic	Model_1	Simplified CNF_1	Model_2	Simplified CNF_2
Initial call	$\{\}$	α	-	-
$SR[B]$	$\{B : True\}$		$\{B : False\}$	

a. -

$$\alpha = (A \lor B \lor C \lor D) \land (\neg A \lor B \lor \neg C) \land (\neg B \lor \neg C \lor D) \land (A \lor \neg B \lor C \lor D)$$
$$\land (B \lor \neg C \lor \neg D) \land (\neg A \lor \neg C \lor D) \land (\neg A \lor \neg D) \land (A \lor \neg B)$$

Table 4:

Heuristic	Model_1	Simplified CNF_1	Model_2	Simplified CNF_2
Initial call	$\{\}$	α	-	-
$SR[B]$	$\{B : True\}$	β	$\{B : False\}$	γ

$$\beta = (\neg C \lor D) \land (A \lor C \lor D) \land (\neg A \lor \neg C \lor D) \land (\neg A \lor \neg D) \land (A)$$
$$\gamma = (A \lor C \lor D) \land (\neg A \lor \neg C) \land (\neg C \lor \neg D) \land (\neg A \lor \neg C \lor D) \land (\neg A \lor \neg D)$$

b. Complete the table below by applying the Unit Clause Heuristic (UCH) as long as possible to the CNF: **[1.5p]**

$\alpha = \neg P \wedge (\neg Q \vee R \vee S) \wedge (\neg P \vee Q) \wedge (P \vee \neg R)$

Use one row for each call and state the variable the heuristic is being applied to in the 1st column, the extension to the model in the 2nd column and the resulting formula in the 3rd column. Add as many additional rows as required.

Table 2:

Heuristic	Model	Simplified CNF
Initial call	$\{\}$	α
$UCH[-]$		
$UCH[-]$		
$UCH[-]$		

b. -

$$\alpha = \neg P \wedge (\neg Q \vee R \vee S) \wedge (\neg P \vee Q) \wedge (P \vee \neg R)$$

Table 5:

Heuristic	Model	Simplified CNF
Initial call	$\{\}$	α
$UCH[\neg P]$	$\{P : False\}$	$(\neg Q \vee R \vee S) \wedge (\neg R)$
$UCH[\neg R]$	$\{P : False, R : False\}$	$(\neg Q \vee S)$

c. Complete the table below by applying the Pure Symbol Heuristic (PSH) as long as possible to the CNF: **[1.5p]**

$\alpha = (\neg P \vee Q) \wedge (R \vee \neg Q) \wedge (M \vee \neg N) \wedge (N \vee \neg M) \wedge (Q \vee N)$

Use one row for each call and state the variable the heuristic is being applied to in the 1st column, the extension to the model in the 2nd column and the resulting formula in the 3rd column. Add as many additional rows as required.

Table 3:

Heuristic	Model	Simplified CNF
Initial call	$\{\}$	α
$PSH[-]$		
$PSH[-]$		
$PSH[-]$		

c. -

$$\alpha = (\neg P \vee Q) \wedge (R \vee \neg Q) \wedge (M \vee \neg N) \wedge (N \vee \neg M) \wedge (Q \vee N)$$

Table 6:

Heuristic	Model	Simplified CNF
Initial call	$\{\}$	α
$PSH[P]$	$\{P : False\}$	$(R \vee \neg Q) \wedge (M \vee \neg N) \wedge (N \vee \neg M) \wedge (Q \vee N)$
$PSH[R]$	$\{P : False, R : True\}$	$(M \vee \neg N) \wedge (N \vee \neg M) \wedge (Q \vee N)$
$PSH[Q]$	$\{P : False, R : True, Q : True\}$	$(M \vee \neg N) \wedge (N \vee \neg M)$

Initially, P, R are pure symbols. Choose one: (P)
R is a pure symbol. Choose one: (R)
Q is a pure symbol. Choose one: (Q)

d. Suppose we had three formulas in propositional logic, $F1, F2$ and $F3$. Explain how one would show whether $F1 \wedge F2 \models F3$ using the DPLL algorithm. In doing this, it is important to show the formal relation between $\models$ and satisfaction. **[2p]**

d. The answer uses the Deduction Theorem and the relationship between satisfiability and validity.

1. The deduction theorem states that If $\Gamma \models \alpha$ then $\models \Gamma \rightarrow \alpha$.
2. $\Gamma \rightarrow \alpha$ is valid iff $\neg(\Gamma \rightarrow \alpha)$ which is the same as saying that $\Gamma \wedge \neg\alpha$ is unsatisfiable.
3. Let β be $F_1 \wedge F_2 \wedge \neg F_3$ in CNF.
4. If DPLL-Satisfiable?(β) is true then $F_1 \wedge F_2 \models F_3$ is false
5. If DPLL-Satisfiable?(β) is false then $F_1 \wedge F_2 \models F_3$ is true

Answer set question [propositionalization or grounding]:

Assume the following signature:

$\Sigma = \langle \mathcal{O} = \{\mathsf{jim}, \mathsf{sarah}\}, \mathcal{P} = \{\mathsf{sameNeighborhood}, \mathsf{strangers}, \mathsf{person}, \mathsf{criminal}, \mathsf{goodNeighbor}\}, \mathcal{V} = \{X, Y\} \rangle$

Propositionalize (ground) the following answer set program Π:

r1: $\mathsf{person}(X)$.

r2: $\mathsf{sameNeighborhood}(X, X)$.

r3: $\mathsf{strangers}(X, Y) \leftarrow \mathit{not}\ \mathsf{sameNeighborhood}(X, Y), \mathsf{person}(X), \mathsf{person}(Y)$.

r4: $\mathsf{goodNeighbor}(X) \leftarrow \mathit{not}\ \mathsf{criminal}(X), \mathsf{person}(X)$.

$Prop(\Pi)$:

r1: person(jim).

r2: person(sarah).

r3: sameNeighborhood(jim, jim).

r4: sameNeighborhood(sarah, sarah).

r5: strangers(jim, jim) $\leftarrow$ *not* sameNeighborhood(jim, jim), person(jim), person(jim).

r6 strangers(sarah, sarah) $\leftarrow$ *not* sameNeighborhood(sarah, sarah), person(sarah), person(sarah).

r7: strangers(jim, sarah) $\leftarrow$ *not* sameNeighborhood(jim, sarah), person(jim), person(sarah).

r8: strangers(sarah, jim) $\leftarrow$ *not* sameNeighborhood(sarah, jim), person(sarah), person(jim).

r9: goodNeighbor(jim) $\leftarrow$ *not* criminal(jim), person(jim).

r10: goodNeighbor(sarah) $\leftarrow$ *not* criminal(sarah), person(sarah).

Question:Answer Sets [Total points: **[8p]**]

The following questions pertain to Answer Set Programming.

Assume the answer set program below Π_1, has the following signature:

$\langle \mathcal{O} = \{a, b\}, \mathcal{P} = \{p, q, r, s\}, \mathcal{V} = \{X\} \rangle$.

Given the positive answer set program Π_1, consisting of the following rules:

r1: $s(a)$.

r2: $p(X) \leftarrow not\ q(X), s(X)$.

r3: $q(X) \leftarrow not\ p(X)$.

r4: $r(X) \leftarrow p(X)$.

r5: $r(X) \leftarrow q(X)$.

a. Propositionalize the answer set program Π_1. Refer to the propositionalized version of Π_1 as Π_p. **[1p]**

b. What is the cardinality of the possible answer sets for Π_p? Explain why. **[1p]**

Given the following *possible* answer sets for Π_p :

$S_1 = \{s(a), q(b), r(b), q(a), r(a)\}$

$S_2 = \{s(a), r(b), q(a), r(a), p(a), p(b)\}$

$S_3 = \{s(a), q(b), r(b), p(a), r(a))\}$

c. For each S_i, provide the reduct, $\Pi_p^{S_i}$ for Π_p. **[1.5p]**

d. Generate $Cn(\Pi_p^{S_i})$ for each reduct $\Pi_1^{S_i}$ of Π_p. **[1.5p]**

e. Which of the three answer sets S_1, S_2, S_3 are *actual* answer sets for Π_p? (Explain why)**[1p]**

f. The consequence relation, $\models$, for classical logic is said to be monotonic. What does this mean? (Be precise and succinct in your explanation) **[1p]**

g. Why is Answer Set Programming considered to be a nonmonotonic reasoning formalism (Be precise and use an example)? **[1p]**

Assume the answer set program below Π_1, has the following signature:

$\langle \mathcal{O} = \{a, b\}, \mathcal{P} = \{p, q, r, s\}, \mathcal{V} = \{X\} \rangle$.

Given the positive answer set program Π_1, consisting of the following rules:

r1: $s(a)$.

r2: $p(X) \leftarrow not\ q(X), s(X)$.

r3: $q(X) \leftarrow not\ p(X)$.

r4: $r(X) \leftarrow p(X)$.

r5: $r(X) \leftarrow q(X)$.

a. Propositionalize the answer set program Π_1. Refer to the propositionalized version of Π_1 as Π_p. **[1p]**

a. -

r1: $s(a)$.

r2: $p(a) \leftarrow not\ q(a), s(a)$.

r3: $q(a) \leftarrow not\ p(a)$.

r4: $r(a) \leftarrow p(a)$.

r5: $r(a) \leftarrow q(a)$.

r6: $p(b) \leftarrow not\ q(b), s(b)$.

r7: $q(b) \leftarrow not\ p(b)$.

r8: $r(b) \leftarrow p(b)$.

r9: $r(b) \leftarrow q(b)$.

b. What is the cardinality of the possible answer sets for Π_p? Explain why. **[1p]**

b. $2^7 = 128$. Seven unique literals in the heads of the rules.

Given the following *possible* answer sets for Π_p :

$$S_1 = \{s(a), q(b), r(b), q(a), r(a)\}$$
$$S_2 = \{s(a), r(b), q(a), r(a), p(a), p(b)\}$$
$$S_3 = \{s(a), q(b), r(b), p(a), r(a))\}$$

a. -

r1: $s(a).$
r2: $p(a) \leftarrow not\ q(a), s(a).$
r3: $q(a) \leftarrow not\ p(a).$
r4: $r(a) \leftarrow p(a).$
r5: $r(a) \leftarrow q(a).$
r6: $p(b) \leftarrow not\ q(b), s(b).$
r7: $q(b) \leftarrow not\ p(b).$
r8: $r(b) \leftarrow p(b).$
r9: $r(b) \leftarrow q(b).$

c. For each S_i, provide the reduct, $\Pi_p^{S_i}$ for Π_p. **[1.5p]**

c. -

$$S_1 = \{s(a), q(b), r(b), q(a), r(a)\}$$
$$S_2 = \{s(a), r(b), q(a), r(a), p(a), p(b)\}$$
$$S_3 = \{s(a), q(b), r(b), p(a), r(a))\}$$

Rule	$\Pi_1^{S_1}$
r1:	$s(a).$
r2:	*delete*
r3:	$q(a).$
r4:	$r(a) \leftarrow p(a).$
r5:	$r(a) \leftarrow q(a).$
r6:	*delete*
r7:	$q(b).$
r8:	$r(b) \leftarrow p(b).$
r9:	$r(b) \leftarrow q(b).$

Rule	$\Pi_1^{S_2}$
r1:	$s(a).$
r2:	*delete.*
r3:	*delete*
r4:	$r(a) \leftarrow p(a).$
r5:	$r(a) \leftarrow q(a).$
r6:	$p(b) \leftarrow s(b).$
r7:	*delete*
r8:	$r(b) \leftarrow p(b).$
r9:	$r(b) \leftarrow q(b).$

Rule	$\Pi_1^{S_3}$
r1:	$s(a).$
r2:	$p(a) \leftarrow s(a).$
r3:	*delete*
r4:	$r(a) \leftarrow p(a).$
r5:	$r(a) \leftarrow q(a).$
r6:	*delete*
r7:	$q(b).$
r8:	$r(b) \leftarrow p(b).$
r9:	$r(b) \leftarrow q(b).$

c. -

$$S_1 = \{s(a), q(b), r(b), q(a), r(a)\}$$
$$S_2 = \{s(a), r(b), q(a), r(a), p(a), p(b)\}$$
$$S_3 = \{s(a), q(b), r(b), p(a), r(a))\}$$

Rule	$\Pi_1^{S_1}$
r1:	$s(a).$
r2:	*delete*
r3:	$q(a).$
r4:	$r(a) \leftarrow p(a).$
r5:	$r(a) \leftarrow q(a).$
r6:	*delete*
r7:	$q(b).$
r8:	$r(b) \leftarrow p(b).$
r9:	$r(b) \leftarrow q(b).$

Rule	$\Pi_1^{S_2}$
r1:	$s(a).$
r2:	*delete.*
r3:	*delete*
r4:	$r(a) \leftarrow p(a).$
r5:	$r(a) \leftarrow q(a).$
r6:	$p(b) \leftarrow s(b).$
r7:	*delete*
r8:	$r(b) \leftarrow p(b).$
r9:	$r(b) \leftarrow q(b).$

Rule	$\Pi_1^{S_3}$
r1:	$s(a).$
r2:	$p(a) \leftarrow s(a).$
r3:	*delete*
r4:	$r(a) \leftarrow p(a).$
r5:	$r(a) \leftarrow q(a).$
r6:	*delete*
r7:	$q(b).$
r8:	$r(b) \leftarrow p(b).$
r9:	$r(b) \leftarrow q(b).$

d. Generate $Cn(\Pi_p^{S_i})$ for each reduct $\Pi_1^{S_i}$ of Π_p. **[1.5p]**

d. -

- $Cn(\Pi_p^{S_1}) : \{\}, \{s(a), q(a), q(b)\}, \{s(a), q(a), q(b), r(a), r(b)\}$
- $Cn(\Pi_p^{S_2}) : \{\}, \{s(a)\}$
- $Cn(\Pi_p^{S_3}) : \{\}, \{s(a), q(b)\}, \{s(a), q(b), p(a), r(b)\}, \{s(a), q(b), p(a), r(b), r(a)\}$

e. Which of the three answer sets S_1, S_2, S_3 are *actual* answer sets for Π_p? (Explain why)**[1p]**

e. $Cn(\Pi_p^{S_1}) = S_1$, $Cn(\Pi_p^{S_2}) \neq S_2$, $Cn(\Pi_p^{S_3}) = S_3$. S_1 and S_3 are answer sets.

f. The consequence relation, $\models$, for classical logic is said to be monotonic. What does this mean? (Be precise and succinct in your explanation) **[1p]**

f. Intuitively, this means that once something is proven relative to a theory, if one extends the theory, anything previously proven will still be proven in the extended theory. This is a form of locality. $\Gamma \models \alpha$ implies that $\Gamma \cup \Gamma' \models \alpha$.

g. Why is Answer Set Programming considered to be a nonmonotonic reasoning formalism (Be precise and use an example)? **[1p]**

g. ASP is considered to be a nonmonotonic formalism, because something inferable from a program Π, may not be inferable when the program is extended with new rules. By inferable, we mean true in all answer sets for the program Π. As an example $\Pi = \{b \leftarrow \mathit{not}\ a\}$ entails b since it has one answer set $\{b\}$. If Π is extended with the rule $a \leftarrow$, then b is no longer entailed since the answer set for the new program is $\{a\}$.