LANGUAGES FOR AI, MODULE 2
DATE 12/02/2026

Time: 2 hours.

You should have three sheets of paper.
Write your name and identification number in the **upper right corner** of ALL the sheets, so that it appears at least once per sheet (not per-page).
Write the answers to exercises 1-2, 3-4 and 5-6 on three different sheets of paper.

You are allowed to use only the following connectives: $\wedge, \vee, \neg, \rightarrow$. **Any other connective will be considered invalid.**

1. (5 points) Consider the language of propositional logic. Use natural deduction to prove that the following holds, or find a counter-example to show that it does not hold (remember that $\neg F$ is only a shorthand for $F \rightarrow \perp$).

   - $\vdash ((\neg A \rightarrow B) \wedge (\neg A \rightarrow \neg B)) \rightarrow A$
   - $\vdash ((A \rightarrow B) \wedge (\neg A \rightarrow C)) \rightarrow (B \wedge C)$

   For students of academic year 2024/25 only: verify for each of the previous formulas whether they are: valid/invalid, satisfiable/unsatisfiable using truth tables. Briefly explain your answer.

2. (5 points) Transform and simplify the following propositional logic formula into an equivalent formula in Disjunctive Normal Form. You are allowed to use only the following connectives: $\wedge, \vee, \neg, \rightarrow$. **Any other connective will be considered invalid.**

$$\neg(A \leftrightarrow B) \rightarrow ((C \rightarrow D) \wedge (D \rightarrow A))$$

3. A rational alarm system is modeled using propositional logic. The propositional variables are defined as follows:

   - A: the alarm is sounding
   - I: there is a real intrusion
   - F: the sensor is faulty
   - R: the rational agent disables the alarm

   The system is described by the following formulas:

   (a) $A \rightarrow (I \vee F)$

   (b) $F \rightarrow \neg I$

   (c) $R \rightarrow \neg A$

   (d) $I \rightarrow A$

   (e) $\neg R$

   Questions.

   1) Is the set of formulas satisfiable?

   2) What happens if we add the formula $A \rightarrow \neg R$ to the theory and we assume that R is true?

4. Consider a FOL language where the domain consists of artificial agents and atomic propositions. The language contains the following predicates:

   - $\mathit{Reliable}(x)$: agent x is reliable
   - $\mathit{Witness}(x, p)$: agent x testifies that the atomic proposition p is true
   - $\mathit{False}(p)$: the atomic proposition p is false

   Assume the following first-order axioms:

(a) **Reliability implies truth**

$$\forall x \forall p(Reliable(x) \land Witness(x,p) \rightarrow \neg False(p))$$

(b) **Existence of a reliable agent**

$$\exists x\ Reliable(x)$$

(c) **Reliable completeness**

$$\forall p\ \exists x(Reliable(x) \land Witness(x,p))$$

(d) **Existence of a false proposition**

$$\exists p\ False(p)$$

(e) **Uniqueness of the reliable agent**

$$\forall x \forall y(Reliable(x) \land Reliable(y) \rightarrow x = y)$$

Questions:

1) Why is the above language First Order, even though we use atomic propositions as arguments of predicates?

2) Prove that the above theory, given by the 5 axioms, is *unsatisfiable* in standard first-order logic.

5. (6 points) A university system manages a list of exam records. Each record can be a valid score (e.g., `exam(subject, grade)`) or a placeholder for a missing exam (e.g., `absent`). Write a *recursive Prolog predicate* `passed_exams(Records, PassedSubjects)` that scans the `Records` list and *constructs a new list* `PassedSubjects`. This new list must contain only the names of subjects with *a grade of 18 or higher*. Records marked as `absent` or with grades lower than 18 must be ignored.

Example Queries:

```
?- passed_exams([exam(math, 28), absent, exam(logic, 15), exam(ai, 30)], L).
L = [math, ai].

?- passed_exams([exam(history, 10), absent], L).
L = [].
```

Write the Prolog code for the `passed_exams/2` predicate below.

6. (6 points) A bakery produces three types of bread bundles: Standard (S), Premium (P), and Deluxe (D). You need to decide how many bundles of each type to pack for a delivery. Write a CLP program to find a valid configuration based on the following constraints:

   (a) **Quantities:** The number of bundles for each type must be between 1 and 50.

   (b) **Total Weight:** The total number of bundles ($S + P + D$) must be exactly **60 bundles**.

   (c) **Cost Constraint:** The cost is different for each type: Standard is 2 euro, Premium is 5 euro, and Deluxe is 10 euro. The `total value` of the load must be exactly `250 euro`.

   (d) **Mix Requirement:** The number of Standard bundles must be strictly greater than the sum of Premium and Deluxe bundles combined ($S > P + D$).

Write the full Prolog/CLP code defining the predicate `bread_load(S, P, D)`.