Question 1. Basic Understanding of KR and Ontologies

In Philosophy, *ontology* is the study of existence and being as such, and of the fundamental classes and relationships of existing things. In Computer Science and Artificial Intelligence, *ontology* is a formal description of knowledge about a domain of interest based on a fixed vocabulary of terms. Explain in 3-5 sentences as to why a logic-based ontology can be used as a "computational" KR model? (3 marks)

Model Solution. Ontologies can be interpreted by computers thanks to a formal (logical) semantics that defines the terms and logical statements using the usual Tarski-style set-theoretic semantics, which enables automated reasoning — the domain is interpreted as a set of elements, an individual as an element in the domain, a class as a subset of the domain, and a relationship as a pair of elements in the domain. This provides human users and computers with a shared understanding of domain knowledge.

Marking Scheme. Award 2 marks when students have, somehow, mentioned the idea of "ontologies are equipped with formal semantics", and unlock the remaining 1 mark when students have (briefly) described the Tarski-style set-theoretic semantics.

Question 2. Expressivity & Computability

Make up a natural language sentence that is unable to be modelled in the formal languages you learnt in the lecture, say in Description Logics or First-Order Logic. Following this example, there may come naturally a belief that a logic-based KR language should be designed as expressive as possible in any circumstances to capture as much domain knowledge as possible. Say in 3-5 sentences your opinion? (3 marks)

Model Solution. A logic-based KR language should be designed as expressive as is able to satisfy the modelling requirements of an application. More expressivity brings more power and flexibility for making statements about domain knowledge. However, on the other hand, such power and flexibility come with a computational cost. The expressive power of the language is invariably constrained so as to at least ensure that reasoning is decidable, i.e., reasoning can always be correctly completed within a finite amount of time.

Marking Scheme. Award 1 mark for a FOL-unmodellable natural language sentence, another 1 mark when students have, somehow, pointed out that there is a trade-off between the expressive power of a language available for making statements and the computational complexity of various reasoning tasks for the language, and unlock the remaining 1 mark when students have mentioned that the expressiveness of a language is due to the fulfillment of the modelling requirements.

Question 3. ALC Extensions & FOL
Consider the following sentences:

- Every Chinese couple have at most 3 children.
- ML is a course taught by ZZH who is a professor working at NJU.
- NJU is a university whose members are a school or a department.
- All members of AI School are undergraduates, graduates, or teachers.

(1) Translate these sentences into one or multiple SHOIQ inclusions. State which concept names, role names, and nominals are used. (4 marks)
(2) Translate the LAST TWO inclusions into equivalent first-order logic. (2 marks)

Model Solutions to (1).

- $\mathsf{Couple} \sqcap \exists \mathsf{hasNationality.Chinese} \sqsubseteq \leq 3 \mathsf{hasChild}.\top$

  Concept names: Couple, Chinese   Role names: hasNationality, hasChild

  An alternative solution:

  $\mathsf{Couple} \sqcap \mathsf{Chinese} \sqsubseteq \leq 3 \mathsf{hasChild}.\top$

  Concept names: Couple, Chinese   Role name: hasChild

- $\{\mathsf{ML}\} \sqsubseteq \mathsf{Course} \sqcap \exists \mathsf{isTaughtBy}.\{\mathsf{ZZH}\}$ (alternative: $\{\mathsf{ML}\} \sqsubseteq \mathsf{Course} \sqcap \exists \mathsf{teaches}^{-}.\{\mathsf{ZZH})\}$
  $\{\mathsf{ZZH}\} \sqsubseteq \exists \mathsf{worksAt}.\{\mathsf{NJU}\}$

  Concept name: Course   Role names: isTaughtBy (teach), workAt   Nominals: {ML}, {ZZH}, {NJU}

- $\{\mathsf{NJU}\} \sqsubseteq \mathsf{University} \sqcap \forall \mathsf{hasMember}.(\mathsf{School} \sqcup \mathsf{Department})$

  Concept names: University, School, Department   Role name: hasMember   Nominal: {NJU}

- $\exists \mathsf{hasMember}^{-}.\{\mathsf{AISchool}\} \sqsubseteq \mathsf{Undergraduate} \sqcup \mathsf{Graduate} \sqcup \mathsf{Teacher}$

  Concept names: Undergraduate, Graduate, Teacher   Role name: hasMember   Nominal: {AISchool}

  An alternative solution:

  $\exists \mathsf{isMemberOf}.\{\mathsf{AISchool}\} \sqsubseteq \mathsf{Undergraduate} \sqcup \mathsf{Graduate} \sqcup \mathsf{Teacher}$

  Concept names: Undergraduate, Graduate, Teacher   Role name: isMemberOf   Nominal: {AISchool}

Model Solutions to (2).

- $\mathsf{University(NJU)} \land \forall y(\mathsf{hasMember(NJU}, y) \rightarrow (\mathsf{School}(y) \lor \mathsf{Department}(y)))$
- $\forall x(\mathsf{hasMember(AISchool}, x) \rightarrow (\mathsf{Undergraduate}(x) \lor \mathsf{Graduate}(x) \lor \mathsf{Teacher}(x)))$

  $\forall x(\mathsf{isMemberOf}(x, \mathsf{AISchool}) \rightarrow (\mathsf{Undergraduate}(x) \lor \mathsf{Graduate}(x) \lor \mathsf{Teacher}(x)))$

Question 4. DL Syntax and Semantics
Consider the interpretation $\mathcal{I}$ defined by

- $\Delta^{\mathcal{I}} = \{a, b, c, d, e\}$
- $P^{\mathcal{I}} = \{a, b, d\}$
- $Q^{\mathcal{I}} = \{d, e\}$
- $r^{\mathcal{I}} = \{(a, b), (a, d), (d, e)\}$

Determine the following sets. (5 marks)

- $(Q \sqcap \geq 2r.P)^{\mathcal{I}}$
- $(\forall r.Q)^{\mathcal{I}}$
- $(\neg \exists r.Q)^{\mathcal{I}}$
- $(\forall r.\top \sqcap \exists r^{-}.P)^{\mathcal{I}}$
- $(\exists r^{-}.\bot)^{\mathcal{I}}$

Model Solutions.

- $(Q \sqcap \geq 2r.P)^{\mathcal{I}} = \emptyset$
- $(\forall r.Q)^{\mathcal{I}} = \{b, c, d, e\}$
- $(\neg \exists r.Q)^{\mathcal{I}} = \{b, c, e\}$
- $(\forall r.\top \sqcap \exists r^{-}.P)^{\mathcal{I}} = \{b, d, e\}$
- $(\exists r^{-}.\bot)^{\mathcal{I}} = \emptyset$

Question 6. Ontology Engineering
Let $\mathcal{T} = \{A \sqsubseteq \exists r.B, B \equiv C, \exists s.B \sqsubseteq C, r \sqsubseteq s, \exists r.C \sqsubseteq C\}$. Determine TWO sets of axioms in $\mathcal{T}$ that are in the pinpointing set Pin($\mathcal{T}$, $A \sqsubseteq C$). (2 marks)

Model Solution.

- $\{A \sqsubseteq \exists r.B, B \equiv C, \exists r.C \sqsubseteq C\}$
- $\{A \sqsubseteq \exists r.B, \exists s.B \sqsubseteq C, r \sqsubseteq s\}$

Question 5. Concept Satisfiability
Consider the ALC-concept C:

$$\neg A \sqcap \neg \forall r.(A \sqcap B) \sqcap \forall r.B$$

Apply the ALC-Tableaux algorithm to C to determine whether C is satisfiable or not. In your answer, show how the completion rules are applied step by step to the constraint system $\{x : C\}$. If C is satisfiable, construct an interpretation $\mathcal{I}$ satisfying C. (7 marks)

Model Solution.

- As C is not in NNF, the first step is to transform C into NNF using the transformation rules. This gives $\neg A \sqcap \exists r.(\neg A \sqcup \neg B) \sqcap \forall r.B$

  Then the ALC-Tableaux algorithm starts with:
  $S_0 = \{x : \neg A \sqcap \exists r.(\neg A \sqcup \neg B) \sqcap \forall r.B\}$

  An application of the $\sqcap$-rule gives:
  $S_1 = S_0 \cup \{x : \neg A, x : \exists r.(\neg A \sqcup \neg B), x : \forall r.B\}$

  An application of the $\exists$-rule gives:
  $S_2 = S_1 \cup \{(x, y) : r, y : \neg A \sqcup \neg B\}$

  An application of the $\sqcup$-rule gives:
  $S_3 = S_2 \cup \{y : \neg A\}$ or $S_3^* = S_2 \cup \{y : \neg B\}$

  An application of the $\forall$-rule gives:
  $S_4^* = S_3^* \cup \{y : B\}$

  Clash obtained, thus we proceed with the other branch:

  An application of the $\forall$-rule gives:
  $S_4 = S_3 \cup \{y : B\}$

  No rule is applicable to S_4 and S_4 contains no clash. Thus, C is *satisfiable.* A model $\mathcal{I}$ of C is given by:

  $\Delta^{\mathcal{I}} = \{a, b, c\}$

  $A^{\mathcal{I}} = \{a\}$

  $B^{\mathcal{I}} = \{b, c\}$

  $r^{\mathcal{I}} = \{(b, b), (b, c)\}$

Question 7. Concept Subsumption via Logical Difference

Recall the definition given in the lecture slides, the logical difference Diff($\mathcal{T}_1$, $\mathcal{T}_2$) from one ontology $\mathcal{T}_1$ to another $\mathcal{T}_2$ are the axioms α entailed by $\mathcal{T}_2$ but not entailed by $\mathcal{T}_1$, i.e., $\mathcal{T}_2 \models \alpha$ and $\mathcal{T}_1 \not\models \alpha$. We call such an axiom a *witness* of Diff($\mathcal{T}_1$, $\mathcal{T}_2$). Consider the following ALC-TBox $\mathcal{T}_1$:

$$\begin{aligned} \mathsf{Bird} &\sqsubseteq \exists \mathsf{hasPart}.\mathsf{Wing} \\ \mathsf{Fish} &\sqsubseteq \exists \mathsf{hasPart}.\mathsf{Fin} \end{aligned}$$

and the following ALC-TBox $\mathcal{T}_2$:

$$\begin{aligned} \mathsf{Bird} &\sqsubseteq \exists \mathsf{hasPart}.\mathsf{Wing} \\ \mathsf{Fish} &\sqsubseteq \exists \mathsf{hasPart}.\top \\ \mathsf{Wing} \sqcap \mathsf{Fin} &\sqsubseteq \bot \end{aligned}$$

(1) Check whether each axiom (one by one) in $\mathcal{T}_2$ is a witness of Diff($\mathcal{T}_1$, $\mathcal{T}_2$) and justify your answers (using Tableaux). (5 marks)

(2) Do the witnesses collected from $\mathcal{T}_2$ make up a complete Diff($\mathcal{T}_1$, $\mathcal{T}_2$)? If yes, justify your answer; if no, give an axiom that is not "explicitly" contained in $\mathcal{T}_2$ but it is a witness of Diff($\mathcal{T}_1$, $\mathcal{T}_2$). (3 marks)

Model Solution to (1).

- Bird $\sqsubseteq$ $\exists$hasPart.Wing is not a witness of Diff($\mathcal{T}_1$, $\mathcal{T}_2$) as it is in $\mathcal{T}_1$ and thus entailed by $\mathcal{T}_1$.
- Fish $\sqsubseteq$ $\exists$hasPart.$\top$ is not a witness of Diff($\mathcal{T}_1$, $\mathcal{T}_2$). One can show this simply by saying that Fish $\sqsubseteq$ $\exists$hasPart.$\top$ is a direct consequence of Fish $\sqsubseteq$ $\exists$hasPart.Fin, or using Tableaux as follows.

  The Tableaux algorithm starts with:
  $S_0 = \{x : \neg\mathsf{Bird} \sqcup \exists\mathsf{hasPart}.\mathsf{Wing}, x : \neg\mathsf{Fish} \sqcup \exists\mathsf{hasPart}.\mathsf{Fin}, x : \mathsf{Fish} \sqcap \forall\mathsf{hasPart}.\bot\}$

  An application of the $\sqcup$-rule gives (branching):
  $S_1 = S_0 \cup \{x : \neg\mathsf{Fish}\}$ or $S_1^* = S_0 \cup \{x : \exists\mathsf{hasPart}.\mathsf{Fin}\}$

  An application of the $\sqcap$-rule gives:
  $S_2 = S_1 \cup \{x : \mathsf{Fish}, x : \forall\mathsf{hasPart}.\bot\}$

  Clash obtained, thus we proceed with the other branch:

  An application of the $\exists$-rule gives:
  $S_2 = S_1^* \cup \{(x, y) : \mathsf{hasPart}, y : \mathsf{Fin}\}$

  An application of the $\sqcap$-rule gives:
  $S_3 = S_2 \cup \{x : \mathsf{Fish}, x : \forall\mathsf{hasPart}.\bot\}$

  An application of the $\forall$-rule gives:
  $S_4 = S_3 \cup \{y : \bot\}$

  Clash obtained, thus Fish $\sqsubseteq$ $\exists$hasPart.$\top$ is entailed by $\mathcal{T}_1$ and NOT a witness of Diff($\mathcal{T}_1$, $\mathcal{T}_2$).
- Wing $\sqcap$ Fin $\sqsubseteq$ $\bot$ is a witness of Diff($\mathcal{T}_1$, $\mathcal{T}_2$). This can be made visible by Tableaux.

  The Tableaux algorithm starts with:
  $S_0 = \{x : \neg\mathsf{Bird} \sqcup \exists\mathsf{hasPart}.\mathsf{Wing}, x : \neg\mathsf{Fish} \sqcup \exists\mathsf{hasPart}.\mathsf{Fin}, x : \mathsf{Fish} \sqcap \mathsf{Fin}\}$

An application of the $\sqcap$-rule gives:
$S_1 = S_0 \cup \{x : \mathsf{Fish}, x : \mathsf{Fin}\}$

An application of the $\sqcup$-rule gives (branching):
$S_2 = S_1 \cup \{x : \neg\mathsf{Fish}\}$ or $S_2^* = S_1 \cup \{x : \exists\mathsf{hasPart.Fin}\}$

Clash obtained in S_2, thus we proceed with S_2^*:

An application of the $\exists$-rule gives:
$S_3 = S_2^* \cup \{(x, y) : \mathsf{hasPart}, y : \mathsf{Fin}\}$

An application of the $\sqcup$-rule gives (branching):
$S_4 = S_3 \cup \{x : \neg\mathsf{Bird}\}$ or $S_4^* = S_3 \cup \{x : \exists\mathsf{hasPart.Wing}\}$

An application of the $\exists$-rule gives:
$S_5 = S_4^* \cup \{(x, z) : \mathsf{hasPart}, z : \mathsf{Wing}\}$

No rule is applicable to S_4 or S_5^*, and neither S_4 nor S_5^* contains clash. Thus, $\mathsf{Wing} \sqcap \mathsf{Fin} \sqsubseteq \bot$ is not entailed by $\mathcal{T}_1$ and a witness of $\mathsf{Diff}(\mathcal{T}_1, \mathcal{T}_2)$.

Model Solution to (2).

They do not make up a complete $\mathsf{Diff}(\mathcal{T}_1, \mathcal{T}_2)$. For example, $\mathsf{Bird} \sqsubseteq \exists\mathsf{hasPart}.\neg\mathsf{Fin}$ is not explicitly contained in $\mathcal{T}_1$, but it is not entailed by $\mathcal{T}_1$ and thus a witness of $\mathsf{Diff}(\mathcal{T}_1, \mathcal{T}_2)$.

Question 8. Concept Subsumption for EL

Consider the following EL-TBox $\mathcal{T}$:

$$\begin{aligned}
\mathsf{Team} &\sqsubseteq \exists \mathsf{hasMember}.\mathsf{Player} \\
\mathsf{VolleyballTeam} &\sqsubseteq \mathsf{Team} \\
\mathsf{Player} &\sqsubseteq \mathsf{Human} \\
\exists \mathsf{hasMember}.\mathsf{Player} &\sqsubseteq \mathsf{Organization}
\end{aligned}$$

(1) Apply the EL-subsumption algorithm given in the lecture slides to compute $S(A)$ for every concept name A in $\mathcal{T}$ and $R(r)$ for every role name r in $\mathcal{T}$. In your answer, show the working steps in the computation of $S(A)$ and $R(r)$. (5 marks)

(2) Using $S(A)$, determine whether VolleyballTeam $\sqsubseteq_{\mathcal{T}}$ Organization. (1 mark)

(3) Using $S(A)$, determine whether Player $\sqsubseteq_{\mathcal{T}}$ Team. (1 mark)

Model Solution to (1).

The initial assignment is given by:

$$\begin{aligned}
S(\mathsf{Team}) &= \{\mathsf{Team}\} \\
S(\mathsf{Player}) &= \{\mathsf{Player}\} \\
S(\mathsf{VolleyballTeam}) &= \{\mathsf{VolleyballTeam}\} \\
S(\mathsf{Human}) &= \{\mathsf{Human}\} \\
S(\mathsf{Organization}) &= \{\mathsf{Organization}\} \\
R(\mathsf{hasMember}) &= \emptyset
\end{aligned}$$

Now applications of (simpleR), (conjR), (leftR), (rightR) are step-by-step as follows:

Update R using (rightR) for the first inclusion of $\mathcal{T}$:
$R(\mathsf{hasMember}) = \{(\mathsf{Team}, \mathsf{Player})\}$

Update S using (simpleR):
$S(\mathsf{VolleyballTeam}) = \{\mathsf{VolleyballTeam}, \mathsf{Team}\}$

Update R using (rightR) for the first inclusion of $\mathcal{T}$:
$R(\mathsf{hasMember}) = \{(\mathsf{Team}, \mathsf{Player}), (\mathsf{VolleyballTeam}, \mathsf{Player})\}$

Update S using (simpleR) for the third inclusion of $\mathcal{T}$:
$S(\mathsf{Player}) = \{\mathsf{Player}, \mathsf{Human}\}$

Update S using (leftR) for the last inclusion of $\mathcal{T}$:
$S(\mathsf{Team}) = \{\mathsf{Team}, \mathsf{Organization}\}$

Update S using (leftR) for the last inclusion of $\mathcal{T}$:
$S(\mathsf{VolleyballTeam}) = \{\mathsf{VolleyballTeam}, \mathsf{Team}, \mathsf{Organization}\}$

The final assignment is:

$$\begin{aligned}
S(\mathsf{Team}) &= \{\mathsf{Team, Organization}\} \\
S(\mathsf{Player}) &= \{\mathsf{Player, Human}\} \\
S(\mathsf{VolleyballTeam}) &= \{\mathsf{VolleyballTeam}, \mathsf{Team}, \mathsf{Organization}\} \\
S(\mathsf{Human}) &= \{\mathsf{Human}\} \\
S(\mathsf{Organization}) &= \{\mathsf{Organization}\} \\
R(\mathsf{hasMember}) &= \{\mathsf{(Team, Player), (VolleyballTeam, Player)}\}
\end{aligned}$$

Model Solution to (2).

Yes.

Model Solution to (3).

No.

Question 9. Ontology-Based Data Access (OBDA)

Consider the following TBox $\mathcal{T}$:

$$\begin{aligned}
\mathsf{Clownfish} &\sqsubseteq \exists\mathsf{hasPart.Fin}\\
\mathsf{Surgeonfish} &\sqsubseteq \exists\mathsf{hasPart.Fin}\\
\exists\mathsf{hasPart.Fin} &\sqsubseteq \mathsf{Fish}\\
\mathsf{Clownfish} \sqcap \mathsf{Surgeonfish} &\sqsubseteq \bot\\
\top &\sqsubseteq \mathsf{Clownfish} \sqcup \mathsf{Surgeonfish}
\end{aligned}$$

and the following ABox $\mathcal{A}$:

$$\begin{aligned}
&\mathsf{Clownfish(nemo)}\\
&\neg\mathsf{Surgeonfish(karl)}\\
&\mathsf{Fish(dory)}\\
&\mathsf{hasPart(dory, krp)}
\end{aligned}$$

Recall that the answers to Boolean queries given by a database instance $\mathcal{I}_{\mathcal{A}}$ corresponding to an ABox $\mathcal{A}$ are "Yes" or "No", and by a knowledge base are "Yes", "No" or "Don't know"

Give the answers in the setting of (i) the database instance $\mathcal{I}_{\mathcal{A}}$ corresponding to the ABox $\mathcal{A}$ (closed world assumption) and (ii) the knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ (open world assumption) to the following Boolean queries and justify your answers (9 marks, 0.5 mark per each for $\mathcal{I}_{\mathcal{A}}$ and 1 mark per each for $\mathcal{K}$):

- $\mathsf{Surgeonfish(nemo)}$
- $\mathsf{Clownfish(karl)}$
- $\mathsf{Fish(karl)}$
- $\mathsf{Fish(krp)}$
- $\mathsf{Fins(krp)}$
- $\exists\mathsf{hasPart.Fins(dory)}$

Model Solutions.

- $\mathsf{Surgeonfish(nemo)}$ No, No
- $\mathsf{Clownfish(karl)}$ No, Yes
- $\mathsf{Fish(karl)}$ No, Yes
- $\mathsf{Fish(krp)}$ No, Yes
- $\mathsf{Fins(krp)}$ No, Yes
- $\exists\mathsf{hasPart.Fins(dory)}$ No, Yes