منطق مرتبه اول

منطق مرتبه اول، منطق مسندها یا محموله‌ای هم نامیده‌می‌شود. علاوه بر ویژگی‌های منطق گزاره‌ای، دارای مفهوم ویژه‌ای به نام پردیکیت(Predicate) یا مسند است.

$$Q\ X\ \ p(t_1, t_2,\ ....\ , t_n)$$

پرسش: مسند یا همان پردیکیت در منطق مرتبه اول، چیست؟ این مفهوم چه قابلیت ویژه به منطق مرتبه اول می‌دهد؟

مثال:

- bird(sparrow)
- partof(engine , car)
- hates(ali , reza)

سورها در منطق مرتبه اول

۱- سور عمومی

∀ X: means for all *X*… (Universal quantification)
Universal quantification ∀ <*variables*> <formula>

Everyone at Stanford is smart
∀X At(X, stanford) ⇒ Smart(X)

∀ X King(X) ⇒ Person(X).
Means that all kings are persons

۲- سور وجودی

∃ X: means for some X. There exists at least one X such that…'(Existential quantification)
Existential quantification ∃ <*variables*> <*sentence*>

Someone at Stanford is smart
∃ X At(X,Stanford) ∧ Smart(X)

∃ X Crown(x) ∧ OnHead(X,John)
Means that something is both a crown and is on the head of John

نکته: مثال‌های بالا برای سورهای عمومی و وجودی، به طور ضمنی به دو نکته مهم اشاره دارد:

نکته اول

Typically, ⇒ is the main connective with ∀
Common mistake: using ∧ as the main connective with ∀
∀x At(x,Stanford) ∧ Smart(x) means "Everyone is at Stanford and everyone is smart"

نکته دوم

Typically, ∧ is the main connective with ∃
Common mistake: using ⇒ as the main connective with ∃:
∃*x* At(x,Stanford) ⇒ Smart(x) is true if there is anyone who is not at Stanford!

گرامر منطق مرتبه اول

$$
\begin{array}{rcl}
Sentence & \rightarrow & AtomicSentence \\
 & | & (\ Sentence\ Connective\ Sentence\ ) \\
 & | & Quantifier\ Variable, \ldots\ Sentence \\
 & | & \neg\ Sentence \\
AtomicSentence & \rightarrow & Predicate(Term, \ldots) \mid Term = Term \\
Term & \rightarrow & Function(Term, \ldots) \\
 & | & Constant \\
 & | & Variable \\
Connective & \rightarrow & \Rightarrow \mid \wedge \mid \vee \mid \Leftrightarrow \\
Quantifier & \rightarrow & \forall \mid \exists \\
Constant & \rightarrow & a \mid x \mid s \mid \cdots \\
Variable & \rightarrow & A \mid X \mid J \mid \cdots \\
Predicate & \rightarrow & Before \mid HasColor \mid Raining \mid \cdots \\
Function & \rightarrow & Mother \mid LeftLeg \mid \cdots
\end{array}
$$

پرسش:

الف – چه شباهت‌ها و تفاوت‌هایی بین گرامر منطق گزاره‌ای و گرامر منطق مرتبه اول وجود دارد؟

ب– مفهوم مسند یا پردیکیت با تابع چه شباهت و تفاوتی دارد؟

پرسش:

آیا جمله $\forall X\ Likes(X, iceCream)$ با جمله $\neg\exists X\ \neg Likes(X,iceCream)$ ، هم ارز است؟

آیا جمله $\exists x\ Likes(X, iceCream)$ با جمله $\neg\forall x\ \neg Likes(X, iceCrem)$ ، هم ارز است؟

نکاتی در مورد سور وجودی و سور عمومی

$$\forall X\ p(X) = \neg\exists X\ \neg p(X)$$
$$\forall X\ \neg p(X) = \neg\exists X\ p(X)$$
$$\exists X\ p(X) = \neg\forall X\ \neg p(X)$$
$$\exists X\ \neg p(X) = \neg\forall X\ p(X)$$

$$\forall X\ p(X) \wedge q = \forall X\,(p(X) \wedge q)$$
$$\forall X\ p(X) \vee q = \forall X\,(p(X) \vee q)$$
$$\exists X\ p(X) \wedge q = \exists X\,(p(X) \wedge q)$$
$$\exists X\ p(X) \vee q = \exists X\,(p(X) \vee q)$$

پرسش: هم ارزی روابط زیر را نشان دهید.

$$\forall X(f(X) \wedge \neg g(X)) = \forall X f(X) \wedge \neg\exists X g(X)$$

$$(False \wedge P) \vee Q = Q$$

$$P \wedge \neg(Q \vee P) = False$$

سورهای تو درتو

- $\forall$ x $\exists$ y Loves(x,y)
  - $\forall$ x ($\exists$ y Loves(x,y)) "everybody loves somebody"
- $\exists$ y $\forall$ x Loves(x,y)
  - $\exists$ y ($\forall$ x Loves(x,y)) "there is someone who is loved by everyone"
- $\exists$ x $\forall$ y Loves(x,y)
  - $\exists$ x ($\forall$ y Loves (x,y)) "there is someone who loves everyone"
- $\forall$ y $\exists$ x Loves(x,y)
  - $\forall$ y ($\exists$ x Loves (x,y)) "everyone is loved by someone

ویژگی‌های دیگر از سورها

- $\forall$x $\forall$y is the same as $\forall$y $\forall$x (typically just write $\forall$ x,y …)$\forall$x $\forall$y Loves(x,y)
  - $\forall$ x ($\forall$ y Loves (x,y)) "everyone loves everyone"
- $\exists$x $\exists$y is the same as $\exists$y $\exists$x (typically just write $\exists$ x,y …)$\exists$x $\exists$y Loves(x,y)
  - $\exists$x ($\exists$y Loves (x,y)) "someone loves someone"

معنای منطق مرتبه اول

تا کنون سورها، رابط‌های منطقی، اتمها، ترمها که عناصر و مولفه‌های گرامری منطق مرتبه اول هستند، توضیح‌داده‌شد. اما برای اینکه این نمادها، زبان را تشکیل دهند، نیاز به تفسیر معنایی دارند تا بازنمایی دانش دنیای پیرامون عامل، شکل بگیرد.

مثال:

- address(ottawa, main, 10) corresponds to a valid Ottawa address,
- onMarket(house(address(ottawa, main, 10), bedrooms(4)), price(225000)) denotes a true or false statement about a specific house.

مفهوم مدل در منطق

فرض‌کنید که S مجموعه‌ای از فرمول‌ها و عبارت‌های منطقی است. اتمها در S ارزش درست یا نادرست دارند. یک مدل از S، زیر مجموعه‌ای از اتمهاست که درست هستند.

مثال:

S1 $\wedge$ S2 is true if both S1 and S2 are in the model.   (This is a <u>hugely</u> simplified approach to models in logic. ☺)

تعریف تاتولوژی با استفاده از مفهوم مدل

A <u>tautology</u> is something that is true in every model. A classic example is this formula true for any H, P:
onMarket(H, P) $\vee \neg$ onMarket(H, P)
Two formulae are <u>model-equivalent</u> if, in every model, they are either both true or both false.

بازتعریف استلزام با استفاده از مفهوم مدل

- **Entailment:** KB $\models$ α whenever every model of KB is also a model of α.
- So KB $\models$ α iff KB$\cup\{\neg$a$\}$ has no model

Formula Φ <u>entails</u> formula Ψ if Ψ is true in every model in which Φ is true. This is written as: $\Phi \models \Psi$

☞ **پرسش:** درستی رابطه زیر را نشان دهید.(راهنمایی: از تعریف استلزام، استفاده کنید)

(onMarket(H, P) $\Rightarrow \neg$ sold(H)) $\wedge$ sold(H) $\models$ $\neg$ onMarket(H, P)

منطق مرتبه اول و پرولوگ

Prolog (PROgramming in LOGic)

پرولوگ (مخفف عبارت «برنامه‌نویسی منطقی» به زبان فرانسوی) یک زبان برنامه‌نویسی است که برای برنامه‌نویسی منطقی به کار می‌رود. در این زبان، برخلاف اکثر زبان‌های دیگر برنامه‌نویسی، به جای دستورالعمل‌های امری، از دانسته‌ها و قواعد منطقی برای حل مسئله استفاده می‌شود. زبان پرولوگ عموماً در حیطه هوش مصنوعی (به ویژه برای پردازش زبان‌های طبیعی) مورد استفاده قرار می‌گیرد.

پرولوگ در سال ۱۹۷۲ در دانشگاه مارسی و توسط Alain Colmerauer و همکاران ابداع شد. پرولوگ مخففی برای "PROgrammation en LOGique" یا «برنامه‌نویسی منطقی» می‌باشد. مستندات اولیه در زمینه پرولوگ، همگی به زبان فرانسه بودند. پس از مدتی در گوشه و کنار دنیا، مخصوصاً در اروپا و ژاپن زبان پرولوگ، طرفدارانی پیدا کرد. گروهی که پرولوگ را ساختند، اساساً یک گروه تحقیقاتی برای پردازش زبان‌های طبیعی برای زبان فرانسه بودند. پرولوگ یک زبان سطح بالا محسوب می‌شود. پرولوگ، زیر مجموعه‌ای از منطق مرتبه اول است.

ساختار برنامه‌های پرولوگ

برنامه‌های پرولوگ از دو مفهوم پایه‌ای زیر تشکیل‌شده‌اند:

Fact

parent(jane, alan). Can be read as “Jane is the parent of Alan.”

Rules

parent(X,Y) :- mother(X,Y). “Person X is the parent of person Y if X is Y’s mother.”

پرسش: مفاهیم پایه‌ای را در برنامه پرولوگ روبرو، تعیین و تفسیر کنید.

```
cat(x) :- furry(x), meows(x), has(x,claws)
furry(Mooshke).
has(Mooshke,claws).

:- cat(Mooshke)
```

پرسش: چرا منطق محموله‌ای، منطق مرتبه اول نامیده می‌شود؟

“First order” means that we can only reason about sets of individuals, that is, only individuals can be quantified. Predicates are fixed. In higher-order logics variables are allowed in predicate positions. Two examples:

مثال:

$\forall X \, \exists Y \, X(Y)$ “each property has at least one instance”

$\forall X \, (X(a) \equiv X(b))$ “a and b have the same properties”

تمرین‌ها

تمرین ۸-۱: هم ارزی روابط زیر را نشان دهید.

$$\neg \exists X \, (f(X) \rightarrow g(X)) \wedge \forall X \neg f(X) = False$$

تمرین ۸-۲: برنامه‌ای به زبان پرولوگ، برای جمع دو عدد طبیعی، ارائه کنید.

باور چیست ؟ از کجا سرچشمه می گیرد؟ هر باور چیزی را حقیقی انگاشتن است. (فردریش نیچه)