# Knowledge Representation and Reasoning

Coursework 2

Daolin Sheng ml192ds

---

## Part A: Sequent Calculus (10 marks)

### Task (i) : $H \wedge Q,\ G\ \vdash\ (\neg H \rightarrow P) \wedge (G \vee R)$

> Answers:
>
> $$\cfrac{\cfrac{\cfrac{\cfrac{Axiom}{H,Q,G \vdash H,P}}{H,Q,G \vdash H \vee P}[\vdash \vee]}{H,Q,G \vdash (\neg H \rightarrow P)}[\vdash \rightarrow r.w.] \quad and \quad \cfrac{\cfrac{Axiom}{H,Q,G \vdash G,R}}{H,Q,G \vdash G \vee R}[\vdash \vee]}{\cfrac{H,Q,G \vdash (\neg H \rightarrow P) \wedge (G \vee R)}{H \wedge Q,G \vdash (\neg H \rightarrow P) \wedge (G \vee R)}[\wedge\ \vdash]}[\vdash\ \wedge]$$

### Task (ii): $\forall x[\neg P(a,x)]\ \vdash\ \forall x[\exists y[\neg P(y,x)]]$

> Answers:
>
> $$\frac{Axiom}{\forall x[\neg p(a,x)], \forall y[p(y,k)], p(a,k) \vdash\ p(a,k)}$$
>
> $$\uparrow$$
>
> $$\frac{\forall x[\neg p(a,x)], \forall y[p(y,k)], p(a,k) \vdash\ p(a,k)}{\forall x[\neg p(a,x)], \forall y[p(y,k)] \vdash\ p(a,k)}[\forall \vdash]$$
>
> $$\uparrow$$
>
> $$\frac{\forall x[\neg p(a,x)], \forall y[p(y,k)] \vdash\ p(a,k)}{\forall x[\neg p(a,x)], \neg p(a,k), \forall y[p(y,k)] \vdash}[\neg \vdash]$$
>
> $$\uparrow$$
>
> $$\frac{\forall x[\neg p(a,x)], \neg p(a,k), \forall y[p(y,k)] \vdash}{\forall x[\neg p(a,x)], \neg p(a,k) \vdash \neg \forall y[p(y,k)]}[\vdash \neg]$$
>
> $$\uparrow$$
>
> $$\frac{\forall x[\neg p(a,x)], \neg p(a,k) \vdash \neg \forall y[p(y,k)]}{\forall x[\neg p(a,x)], \neg p(a,k)\ \vdash\ \exists y[\neg p(y,k)]}[\vdash \exists r.w.]$$
>
> $$\uparrow$$

$$\dfrac{\forall x[\neg p(a,x)], \neg p(a,k) \ \vdash \ \exists y[\neg p(y,k)]}{\forall x[\neg P(a,x)], \neg p(a,k) \ \vdash \ \forall x[\exists y[\neg P(y,x)]]} \ [\vdash \forall]^{\dagger}$$

$$\uparrow$$

$$\dfrac{\forall x[\neg P(a,x)], \neg p(a,k) \ \vdash \ \forall x[\exists y[\neg P(y,x)]]}{\forall x[\neg P(a,x)] \ \vdash \ \forall x[\exists y[\neg P(y,x)]]} \ [\forall \ \vdash]$$

## Part B: Axioms and Models (20 Marks)

### Task (i)

Answers:

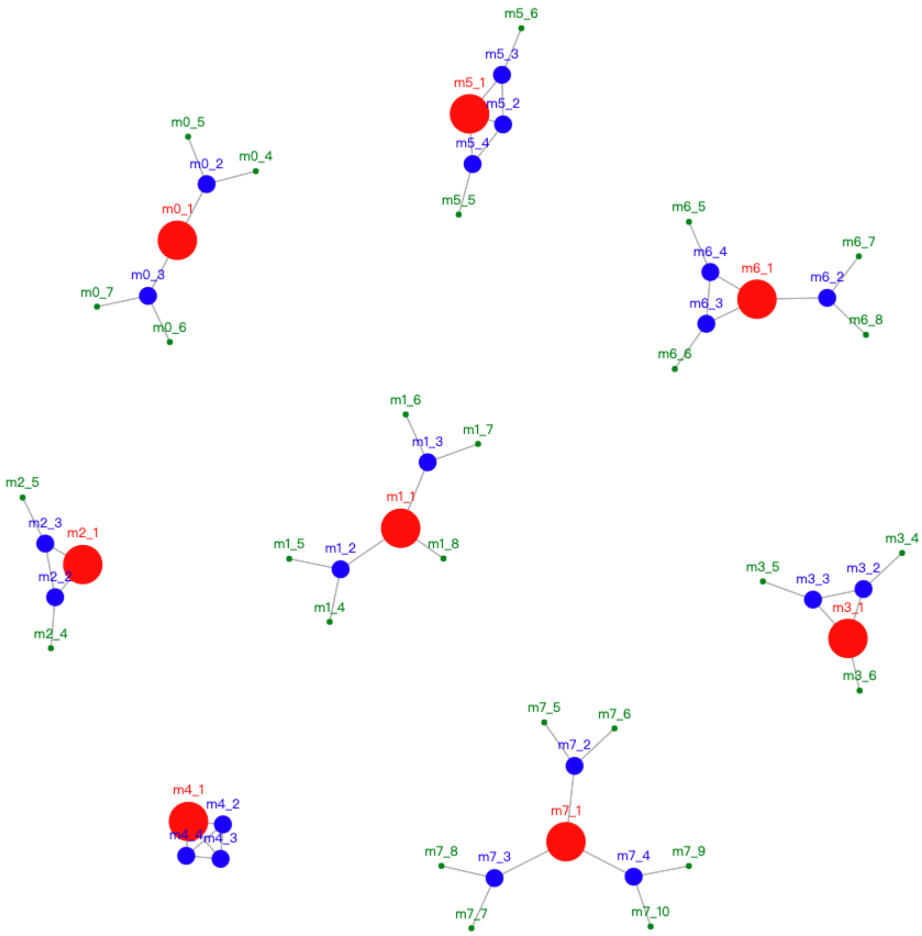
1. If there is a road between A place and B place, then there is also a road between B place and A place and A and B are different places.
2. If Any A place is larger than any B place, then only when A is a city and B is a town or village or A is a town and B is a village.
3. Every place is a city or town or village.
4. There is at least one city in this area.
5. There is at most one city in this area.
6. This area has at least two towns.
7. Every place except cities has a road to a larger place.
8. Every village can be connected to at most one place.
9. Every town can be connected to at least three different places.
10. Every place can be connected to at most three different places.

### Task (ii)

Answers:

- Submit the all models in Python file by Gradescope.

- Draw out the graphical representation of models following:




## Part C: Build Your Own Knowledge Base (30 marks)

### Initial Experimentation

1. rule(hate, [[X, is, cat], [Y, is, dog], -[Y, is, puppy]] ==> [X, hates, Y])
2. rule(family, [[X, is_father_of, Y], [Y, is_father_of,Z]] ==> [X, is_grandfather_of,Z]).
3. rule(genesis, [[X, is, dog]] ==> [fatherof(X), is, dog] ).
   - The purpose of this rule is that if X is a dog, then X's father is a dog.
   - This rule does not works because fatherof() is not exits.
4. When we use the weak operator +, it will regard the uncertain conditions as True.

### Task(i): Your Knowledge Base

> Answers:
>
> - The code have submitted as separate file on Gradescope.
> - The code also can see by this web address: https://swish.swi-prolog.org/p/ml192ds_kb.pl

### Task(ii): Discussion of your KB and its Inferences

**(1).** My knowledge base is base on six students who comes from different countries and have different ages, gender and are good at different skills. They registered different modules which are knowledge representation and reasoning(krr), machine learning(ml), programming for data science(pds), scientific computation(sc) and advanced software engineering(ase), and these modules teach by four teachers. Besides, every modules requires different skills, such as math, logic, writing or coding and also teach different programming languages that students can be learn. Finally, I suppose the time that different students spend on different modules each week is different and store in a list.

By setting different facts and rules, it can inference they have common country if they come the same country. As the same time, if they have same module and come from the same country, as a result, they are can make a study group. Besides, because every modules requires different skills and every students is good at different skills and studying time on every module per week is different, we can inference he/she would get a good performance if he/she gives enough time on a module. Then, the teacher who teach a module will be happy for who has a good performance on this module.

The limitation of this Knowledge Base is that "likes" in the rule 'rule( default, [[X, and, Y, have_common_interest], +(-[X, likes, Y]) ] ==> [X, likes, Y] ).' can cause ambiguity. By running 'infer(3), describe([likes])', for instance, it output [billy, likes, cola] and [billy, likes, logic], in this case, the 'likes' indicates liking a people and a skill.

**(2).**

1. "About the Billy?", by running 'infer(1), describe([billy])', it can inference out that Billy comes from china, 25 years old has common country with Alice and has common skill with cola, and Billy can make friend with cola.
2. "Who have the common interests?", by running 'infer(2), describe(have_common_interest)', it can be inference out Cola likes Billy, and Cola has common interest with Dota.
3. "Who are best student?", by running '?- infer(3), describe([happy_for, is_best_student]).', it can be inference out Billy and Dota have good performance and are the best student and Brandon and Matteo are happy for them.