Started on	Wednesday, 7 April 2021, 3:07 PM
State	Finished
Completed on	Wednesday, 7 April 2021, 5:02 PM
Time taken	1 hour 55 mins
Grade	Not yet graded

Question **1**

Complete

Not graded

Legal Statement:

I hereby confirm that:

1. I'm the person that is enrolled in the course of "Knowledge Representation and Engineering" and whom the following test is intended to.
2. During the time of the test, I won't make use of any external documentation or will receive help of third persons.
3. I won't use any means to overcome the security measures implemented in the evaluation tool (moodle).
4. I accept that, if any attempt to breach the above points or any other that intentionally represents a fraudulent use of the evaluation process is detected, the Universitat Rovira i Virgili may undertake the established disciplinary measures.

Select one:

- (•) True
- ( ) False

The correct answer is 'True'.

Question **2**

Complete

Marked out of 2.00

Isolate the **units of knowledge** of the following paragraph and identify the **declarative knowledge** and the **procedural knowledge**:

"*A chemical bond is a lasting attraction between atoms, ions or molecules that enables the formation of chemical compounds. The bond may result from the electrostatic force of attraction between oppositely charged ions as in ionic bonds; or through the sharing of electrons as in covalent bonds*".

A chemical bond is a lasting attraction between atoms,ions or molecules that enables the formation of chemical compounds.
+ Knowledge :: A chemical bound is a lasting attraction
+ Knowledge :: A chemical bound involves atoms, ions or molecules
+ Knowledge :: A chemical bound enables the formation of chemical compounds
+ Know-what :: This is a fact

The bond may result from the electrostatic force of attraction between oppositely charged ions as in ionic bonds; or through the sharing of electrons as in covalent bonds
+ Knowledge :: A chemical bond can be caused by force of attraction between oppositely charged ions
+ Knowledge :: Ionic bonds form by electrostatic force of attraction between oppositely charged ions
+ Knowledge :: Ionic bonds are chemical bonds
+ Knowledge :: A chemical bond the sharing of electrons
+ Knowledge :: Covalent bonds form by sharing of electrons
+ Knowledge :: Covalent bonds are chemical bonds
+ Know-what :: This a fact

The units of knowledge are:

U1: "A chemical bond is a lasting attraction between atoms, ions or molecules"

U2: "[a chemical bond] enables the formation of chemical compounds"

U3: "Ionic bonds and covalent bonds are two kinds of chemical bonds"

U4: "Ionic bond results from the electrostatic force of attraction between oppositely charged ions"

U5: "Covalen bond results from the sharing of electrons"

Among these units of knowledge U1, U2, U3 is declararive (know-what) and U4, U5 procedural (know-why).

Question **3**

Complete

Marked out of 2.00

In **First Order Logic** notation, represent the following knowledge about a pet care service:

1. *Among all animals, only dogs and cats are accepted.*
2. *Dogs and cats are kept in separated rooms.*
3. *Dogs are walked twice a day in groups of three.*

Copy the following symbols and Paste them within your statements, if required: ∀ ∃ ∧ ¬ ∨ ⊃

1. ∀x Accepted(x) ⊃ Dog(x) ∨ Cat(x)
2. ∀x∀y∀r1∀r2 Dog(x) ∧ Cat(y) ∧ Room(r1) ∧ Room(r2) ∧ in(x,r1) ∧ in(y,r2) ⊃ r1≠r2
3. (∀x Dog(x) ⊃ walksPerDay(x) = 3) ∧ (∀x∀w Dog(x) ∧ DailyWalk(w) ∧ in(x,w) ⊃ ∃y∃z in(y,w) ∧in(z,w))

1. ∀x. animal(x) ∧ accepted(x) ⊃ dog(x) ∨ cat(x)
2. ∀x.∀y.∀r1.∀r2. dog(x) ∧ cat(y) ∧ in(x, r1) ∧ in(x, r2) ⊃ r1 ≠ r2
3. ∀x. dog(x) ⊃ walked(x, 2) ∧ (∃y.∃z. dog(y) ∧ dog(z) ∧ x≠y ∧ y≠z ∧ x≠z ∧ walked_with(x,y) ∧ walked_with(y,z))

Question **4**

Complete

Marked out of 2.00

Use the **rule notation** to represent the knowledge about the 14th Article of the Spanish Constitution, when it states that:

"*Spaniards are equal before the law, and no discrimination based on birth, race, sex, religion, opinion or any other personal or social condition or circumstance that may prevail.*"

Copy the following symbols and Paste them within your statements, if required: ∧ ∨€¬ ⇐

Factor(*birth)* ∨ *Factor(race)* ∨ *Factor(sex)* ∨ *Factor(religion)* ∨ *Factor(opinion)* ∨ *Factor(condition)* ∨ *Factor(circumstance)*
isLaw(l,x) <= Spaniard(x) ∧ Law(l)
¯prevailOn(x,y) <= Spaniard(x) ∧ Law(l) ∧ Factor(y) ∧ FactorOf(x, y)

1. "*Spaniards are equal before the law*"

€applies(l,y) ⇐€spaniard(x) ∧ spaniard(y) ∧ applies(l,x) ∧ law(l)

2. "*birth, race, sex, religion, opinion or any other personal or social condition or circumstance*"

This defines a hierarchical relationship between birth, race, sex, religion, opinion as personal or social conditions and these as condition-circumstance (these conditions or circumstances refer to people):

personal_condition(b,x) ⇐€birth(x,b)

personal_condition(r,x) ⇐€race(x,r)

personal_condition(a,x) ⇐€sex(x,s)

social_condition(r,x) ⇐€religion(x,r)

personal_condition(o,x) ⇐€opinion(x,o)

condition_circumstance(c,x) ⇐€personal_condition(c,x)

Question **5**

Complete

Marked out of 2.00

Use a rule production system to represent the knowledge about the operation of a smart elevator described as:

"The elevator is always in one position (floor) going to a destination, moving or stopped and carrying a number of passengers **(elevator position:{1..N} destination:{1..n} status:{moving, stopped} passengers:integer)**. Users arrive at a floor willing to go to another floor and they can be waiting or in the elevator **(user id:number arrives:{1..n} goes-to:{1..N} status:{in, waiting})**. When the elevator is empty and there are not user calls, it goes to the main floor. When stopped in a floor without waiting users, empty elevator attends the user call of the highest upper floor. When the elevator arrives to a floor it stops, if it carries some user to this floor or if there's some user waiting in this floor. If the elevator is stopped in one floor, first it lets the users in the elevator going to that floor to leave, and then the users waiting in that floor get in the elevator. If the elevator is moving and arrives to one floor where nobody has to get off and nobody is waiting outside, the elevator continues to the following floor. "

1. When the elevator is empty and there are not user calls, it goes to the main floor.

IF (elevator passengers:{=0} position:{>0}) -(user) THEN MODIFY 1 (destination 0) (status moving)

2. When stopped in a floor without waiting users, empty elevator attends the user call of the highest upper floor.

IF (elevator position:p status:{=stopped} passengers:{=0}) -(user arrives:p) (user arrives:q) -(user arrives:{>q}) THEN MODIFY 1 (destination q) (status [moving])

3. When the elevator arrives to a floor it stops, if it carries some user to this floor or if there's some user waiting in this floor.

IF (elevator position:p status:{=moving}) (user goes-to:p status:{=in}) THEN MODIFY 1 (status [stopped])

IF (elevator position:p status:{=moving}) (user arrives:p status:{=waiting}) THEN MODIFY 1 (status [stopped])

4. If the elevator is stopped in one floor, first it lets the users in the elevator going to that floor to leave, and then the users waiting in that floor get in the elevator.

IF (elevator position:p status:{=stopped} passengers:n) (user goes-to:p status:{=in}) THEN MODIFY 1 (passengers [n-1]) REMOVE 2

IF (elevator position:p status:{=stopped} passengers:n) -(user goes-to:p status:{=in}) (user arrives:p status:{=waiting}) THEN MODIFY 1 (passengers [n+1]) MODIFY 3 (status [in])

5. If the elevator is moving and arrives to one floor where nobody has to get off and nobody is waiting outside, the elevator continues to the following floor.

IF (elevator position:p destination:{>p} status:{=moving}) -(user goes-to:p status:{=in}) -(user arrives:p status:{=waiting}) THEN MODIFY 1 (position [p+1]) ---moves up

IF (elevator position:p destination:{<p} status:{=moving}) -(user goes-to:p status:{=in}) -(user arrives:p status:{=waiting}) THEN MODIFY 1 (position [p-1]) ---moves down

Question **6**

Complete

Marked out of 2.00

All flights have a destination and a capacity for passengers, but more tickets than capacity can be sold (overbooking). During check-in (admission of passenger to a plane), a passenger who doesn't fit into the plane must be checked-in to an alternative flight with the same destination which has free seats at the time of that particular check-in. Propose a system of frames that allows check-in and flight reassignment, according to this procedure.

Flight frame will have slots for destination and capacity (that decreases as passengers check-in):

```
(Flight
  <:IS-A Thing>
  <:Destination Airport>
  <:Capacity number>
)
```

Passengers are assigned to flights during check-in:

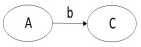
```
(Passenger
  <:IS-A Thing>
  <:Flight Flight
     [IF-ADDED {
          (lambda (flight)
             (if (> flight:Capacity 0)
                (progn (set flight:Capacity (+1 flight:Capacity)) flight)
                (set :SELF:Flight (find-next-flight flight:Destination)))))}]>
)
```

Question **7**

Complete

Marked out of 2.00

Propose a definitional semantic network to represent the following facts:

NOTE: use the textual notation **b(A, C)** to represent the semantic network relationship 

a) "*All the birds are animals*."

b) "*Not all the birds can fly*."

c) "*The beak is an external anatomical structure of birds that is used for eating, preening, and fighting*."

d) "*Woody is a woodpecker, and woodpeckers use their beak to make holes in the wood*."

a) aKindOf(Bird, Animal)

b) can(Bird,Fly)

c) aKindOf(Beak,AnatomicalStructure), aPartOf(Beak, Bird), usedTo(Beak, Eat), usedTo(Beak, Preen), usedTo(Beak, Fight)

d) isA(Woody,Woodpecker), aKindOf(Woodpecker, Bird), usedTo(Beak, MakeHoles),

a) ako(Bird, Animal) --- ako== a kind of

b) ako(Flying Bird, Bird); ako(Other Bird, Bird); can(Flying Bird, Fly)

c) ako(Beak, External Anatomical Structure); ako(External Anatomical Structure, Bird Anatomical Structure); part-of(Bird Anatomical Structure, Bird); part-of(Beak, Bird); used-for(Beak, Beak Use)

... if we take the uses of the beak as an AND: ako(Beak Use, Eating); ako(Beak Use, Preening), ako(Beak Use, Fighting)

... if we take the uses of the beak as an OR: ako(Eating, Beak Use), ako(Preening, Beak Use), ako(Fighting, Beak Use)

d) is-a(Woody, Woodpecker); part-of(Woodpecker Beak, Woodpecker); ako(Woodpecker Beak, Beak), used-for(Woodpecker Beak, WP Beak Use); action(WP Beak Use, Making Holes); place(WP Beak Use, Wood)

Note: other possible representations are possible.

◄ Guidelines for On-line Exams

Jump to...

Submit PR1 ►