Οικονομικό Πανεπιστήμιο Αθηνών, Τμήμα Πληροφορικής
Μάθημα: Τεχνητή Νοημοσύνη, 2024–25
Διδάσκων: Ι. Ανδρουτσόπουλος

Οι διαφάνειες και οι ασκήσεις μελέτης της 13ης διάλεξης παρέχονται ως προαιρετικό υλικό μελέτης. Δεν θα καλυφθούν από τις διαλέξεις και τα φροντιστήρια και δεν περιλαμβάνονται στην εξεταστέα ύλη του 2024–25.

Ασκήσεις μελέτης της 13ης διάλεξης

13.1. Παραστήστε σε OWL με συντακτικό συναρτησιακού ύφους (functional-style syntax) τις σημασίες των προτάσεων (i) – (v). Ακολουθούν πρόσθετες πληροφορίες και υποδείξεις:

- Η τάξη των αντικειμένων που έχουν το :psita ως τιμή της ιδιότητας (property) :likes παριστάνεται ως ObjectHasValue(:likes :psita)
- Η ένωση δύο τάξεων C_1 και C_2 παριστάνεται ως ObjectUnionOf(C_1 C_2).
- Η τομή δύο τάξεων C_1 και C_2 παριστάνεται ως ObjectIntersectionOf(C_1 C_2).
- Η τάξη των αντικειμένων που έχουν ως τιμή της ιδιότητας :likes *τουλάχιστον ένα* αντικείμενο της τάξης :Cat παριστάνεται ως ObjectMinCardinality(1 :likes :Cat).

(i) Ο Μίλος είναι σκύλος και η Ψίτα είναι γάτα.

```
ClassAssertion( :Dog :milos )
ClassAssertion( :Cat :psita )
```

(ii) Κάθε σκύλος γαβγίζει ή κουνάει την ουρά του (ενδεχομένως και τα δύο μαζί).

```
SubClassOf( :Dog
            ObjectUnionOf( :Barks :MovesTail ))
```

(iii) Κάθε σκύλος που συμπαθεί την Ψίτα γαβγίζει.

```
SubClassOf( ObjectIntersectionOf( :Dog
                                  ObjectHasValue( :likes :psita ))
            :Barks )
```

(iv) Ο Μίλος συμπαθεί την Ψίτα.

```
ObjectPropertyAssertion( :likes :milos :psita )
```

(v) Κάθε σκύλος που συμπαθεί μια (τουλάχιστον) γάτα γαβγίζει.

```
SubClassOf( ObjectIntersectionOf( :Dog
                                  ObjectMinCardinality( 1 :likes Cat ))
            :Barks)
```

13.2. Παραστήστε σε OWL (με functional-style syntax) τις προτάσεις (i)–(vii). Επιτρέπεται να χρησιμοποιήσετε μόνο τα ακόλουθα ονόματα τάξεων: *Human*, *Woman*. Επίσης, επιτρέπεται να χρησιμοποιήσετε μόνο τα ακόλουθα ονόματα ιδιοτήτων (properties): *HasGrandSon*, *HasChild*, *HasBrother*, *HasMother*.

(i) Η Μαρία είναι γυναίκα. *ClassAssertion(:Woman :Maria)*

(ii) Η Μαρία έχει το Νίκο εγγόνι. *ObjectPropertyAssertion(:hasGrandChild :Maria :Nikos)*

(iii) Η Μαρία έχει τουλάχιστον ένα παιδί.

```
ClassAssertion( ObjectMinCardinality( 1 :hasChild )
                :Maria )
```

(iv) Ο Γιάννης έχει τουλάχιστον έναν αδελφό.

```
ClassAssertion( ObjectMinCardinality( 1 :hasBrother )
                :John )
```

(v) Ο Γιάννης έχει **ακριβώς** έναν αδελφό.

```
ClassAssertion( ObjectExactCardinality( 1 :hasBrother )
                :John )
```

(vi) Κάθε άνθρωπος έχει **ακριβώς** μία μητέρα.

```
SubClassOf( :Human
            ObjectExactCardinality( 1 :hasMother ))
```

(vii) Αν μία γυναίκα έχει τουλάχιστον ένα εγγόνι, τότε έχει και τουλάχιστον ένα παιδί.

```
SubClassOf( ObjectIntersectionOf( :Woman
                                  ObjectMinCardinality( 1 :hasGrandChild ))
            ObjectMinCardinality( 1 :hasChild ))
```

13.3. Ορίστε σε OWL μια οντολογία που να περιλαμβάνει τις ακόλουθες τάξεις, με τις σημασίες που υπονοούν τα ονόματά τους:

Human, Man, Woman

και τις ακόλουθες σχέσεις, με τις σημασίες που υπονοούν τα ονόματά τους:

isHusbandOf, isWifeOf,
isFatherOf, isMotherOf, isSonOf, isDaughterOf, isParentOf, isChildOf,
isGrandFatherOf, isGrandMotherOf, isGrandSonOf, isGrandDaughterOf.

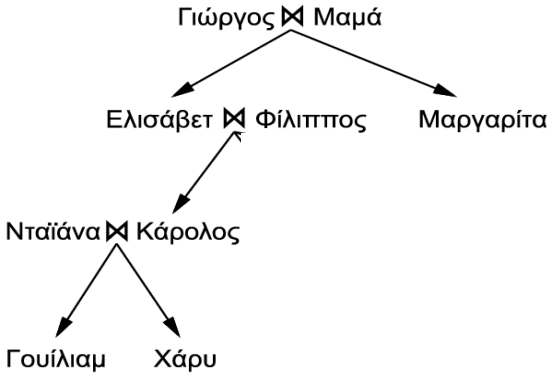
Η οντολογία σας πρέπει να περιλαμβάνει τις λιγότερες δυνατές δηλώσεις, αλλά να καλύπτει όλους τους εύλογους περιορισμούς, οι οποίοι να αποκλείουν π.χ. να δηλωθεί ως πατέρας κάποιου μία γυναίκα ή να έχει κάποιος περισσότερες από μία (βιολογικές) μητέρες κλπ. Η οντολογία σας πρέπει, επίσης, να υποστηρίζει την αυτόματη εξαγωγή εύλογων συμπερασμάτων, όπως π.χ. ότι αν ο/η Χ είναι γονέας του/της Υ, τότε ο/η Υ είναι παιδί του/της Χ.

Μελετήστε πρώτα το «OWL 2 Primer» (βλ. http://www.w3.org/TR/owl2-primer/). Χρησιμοποιήστε «functional-style syntax». Ενδέχεται να χρειαστεί να χρησιμοποιήσετε υπο-ιδιότητες (sub-properties) και αλυσίδες ιδιοτήτων (property chains). Υπόδειξη: Για κάθε είδος δήλωσης που περιγράφει το «OWL 2 Primer», σκεφτείτε προσεκτικά μήπως πρέπει να προσθέσετε κάποια αντίστοιχη δήλωση στην οντολογία της άσκησης.

Απάντηση:

```
/* Class Definitions */
SubClassOf(:Man :Human)                                    //Man ISA Human
SubClassOf(:Woman :Human)                                  //Woman ISA Human
DisjointClasses(:Man :Woman)                               //Man != Woman
EquivalentClasses(:Human ObjectUnionOf(:Man :Woman))       //Man + Woman = Human

/* Property Definitions: Husband – Wife */
ObjectPropertyDomain(:isHusbandOf :Man)          //isHusbandOf's domain = Man
ObjectPropertyRange(:isHusbandOf :Woman)         //isHusbandOf's range = Woman
FunctionalObjectProperty(:isHusbandOf)    //a Man can be the husband of only one Woman
FunctionalObjectProperty(:isWifeOf)          //a Woman can be the wife of only one Man
InverseObjectProperties(:isHusbandOf :isWifeOf)  //if x is the husband of y, y is the wife of x

/* Property Definitions: Parent – Child */
ObjectPropertyDomain(:isParentOf :Human)      //isParentOf's domain = Human
ObjectPropertyDomain(:isChildOf :Human)       //isChildOf's domain = Human
IrreflexiveObjectProperty(:isParentOf)        // x can't be the parent of x
AsymmetricObjectProperty(:isParentOf)         // if x is a parent of y, y can't be a parent of x
InverseObjectProperties(:isParentOf :isChildOf) //if x is a parent of y, then y is a child of x
SubObjectPropertyOf(ObjectPropertyChain(:isChildOf :isHusbandOf) :isChildOf)
        //if x is a child of y and y is the husband of z, then x is a child of z
SubObjectPropertyOf(ObjectPropertyChain(:isChildOf :isWifeOf) :isChildOf)
        //if x is a child of y and y is the wife of z, then x is a child of z
SubClassOf(:Human :ObjectExactCardinality(1 :isChildOf :Man))
        //a Human is the child of exactly one Man
SubClassOf(:Human :ObjectExactCardinality(1 :isChildOf :Woman))
        //a Human is the child of exactly one Woman

/* Property Definitions: Father – Mother */
SubObjectPropertyOf(:isFatherOf :isParentOf)   //if x is the father of y, then x is a parent of y
SubObjectPropertyOf(:isMotherOf :isParentOf)   //if x is the mother of y, x is a parent of y
ObjectPropertyDomain(:isFatherOf :Man)         //isFatherOf's domain = Man
ObjectPropertyDomain(:isMotherOf :Woman)       //isMotherOf's domain = Woman

/* Property Definitions: Son – Daughter */
SubObjectPropertyOf(:isSonOf :isChildOf)       //if x is a son of y, then x is a child of y
SubObjectPropertyOf(:isDaughterOf :isChildOf)  //if x is a daughter of y, then x is a child of y
ObjectPropertyDomain(:isSonOf :Man)            //isSonOf's domain = Man
ObjectPropertyDomain(:isDaughterOf :Woman) //isDaughterOf's domain = Woman

/* Property Definitions: GrandFather – GrandMother – GrandSon – GrandDaughter */
SubObjectPropertyOf(ObjectPropertyChain(:isFatherOf :isParentOf) :isGrandFatherOf)
        //if x is the father of y and y is a parent of z then x is a grand-father of z
SubObjectPropertyOf (ObjectPropertyChain(:isMotherOf :isParentOf) :isGrandMotherOf)
        //if x is the mother of y and y is a parent of z then x is a grand-mother of z
SubObjectPropertyOf (ObjectPropertyChain(:isSonOf :isChildOf) :isGrandSonOf)
        //if x is a son of y and y is a child of z then x is grand-son of z
SubObjectPropertyOf (ObjectPropertyChain(:isDaughterOf :isChildOf) :isGrandDaughterOf)
        //if x is a daughter of y and y is a child of z then x is grand-daughter of z
```

13.4. Χρησιμοποιήστε την οντολογία της προηγούμενης άσκησης για να περιγράψετε πλήρως σε OWL τις οικογενειακές πληροφορίες του παρακάτω σχήματος με όσο το δυνατόν λιγότερες δηλώσεις.[1] Δεν χρειάζεται να γράψετε δηλώσεις differentIndividuals.



Απάντηση:

```
ObjectPropertyAssertion(:isHusbandOf :Γιώργος :Μαμά)
ObjectPropertyAssertion(:isDaughterOf :Ελισάβετ :Γιώργος)
ObjectPropertyAssertion(:isDaughterOf :Μαργαρίτα :Γιώργος)
ObjectPropertyAssertion(:isHusbandOf :Φίλιππος :Ελισάβετ)
ObjectPropertyAssertion(:isSonOf :Κάρολος :Φίλιππος)
ObjectPropertyAssertion(:isHusbandOf :Κάρολος :Νταϊάνα)
ObjectPropertyAssertion(:isSonOf :Γουίλιαμ :Κάρολος)
ObjectPropertyAssertion(:isSonOf :Χάρυ :Κάρολος)
```

[1] Τροποποιημένη μορφή σχήματος των διαφανειών του βιβλίου «Artificial Intelligence – A Modern Approach» των S. Russel και P. Norvig, 2η έκδοση, Prentice Hall, 2003.