
ΚΕΦΑΛΑΙΟ 22

Τεχνολογία Γνώσης

Ασκήσεις - Ερωτήσεις

1. Έστω ότι κάποιος θέλει να αγοράσει ένα αυτοκίνητο. Αυτού του είδους τα προβλήματα δεν είναι καλά ορισμένα, γιατί υπάρχει υποκειμενικότητα. Παρόλα αυτά το πρόβλημα της επιλογής μπορεί να αντιμετωπιστεί με ένα σύστημα γνώσης, αν οριστούν κάποιες παράμετροι στο πρόβλημα, όπως για παράδειγμα πόσα χρήματα είναι διαθέσιμα, για ποιο σκοπό είναι απαραίτητη η χρήση του αυτοκινήτου, κατασκευαστές και μοντέλα τα οποία απορρίπτει εξαρχής ο υποψήφιος αγοραστής, κτλ.
 - α) Να καταγραφούν οι πιο σημαντικές έννοιες του προβλήματος καθώς και οι συσχετίσεις μεταξύ τους, όπως κατασκευαστές και μοντέλα αυτοκινήτων, τα χαρακτηριστικά τους, για παράδειγμα κυβισμός, ιπποδύναμη, κατανάλωση. Επίσης, τα παραπάνω μπορεί να συσχετιστούν με σημαντικούς παράγοντες που χαρακτηρίζουν τον υποψήφιο αγοραστή, όπως πόσες ώρες οδηγεί την ημέρα, αν έχει σκάφος ή τροχόσπιτο, κτλ.
 - β) Να αναπαρασταθούν τα παραπάνω με κάποιες από τις τυπικές μορφές αναπαράστασης γνώσης. Για παράδειγμα, τα μοντέλα των αυτοκινήτων μπορεί να αναπαρασταθούν με πλαίσια, τα οποία μπορεί να είναι ομαδοποιημένα σε κλάσεις, είτε ανά κατασκευαστή ή ανά τύπο αυτοκινήτου, για παράδειγμα σπορ, οικογενειακό, επαγγελματικό. Είναι δυνατό να υπάρχουν περισσότερες της μίας διαστάσεις βάσει των οποίων μπορεί να γίνει η κατηγοριοποίηση των αυτοκινήτων, επομένως μπορεί να χρησιμοποιηθεί πολλαπλή κληρονομικότητα (multiple inheritance).
 - γ) Ένα σημαντικό σημείο για την κατασκευή ενός συστήματος γνώσης για επιλογή αυτοκινήτου είναι η ιεράρχηση των προτεραιοτήτων όσον αφορά τα επιθυμητά χαρακτηριστικά καθώς και η επίλυση συγκρούσεων μεταξύ αντικρουόμενων ιδιοτήτων. Για παράδειγμα, αν ο υποψήφιος αγοραστής επιθυμεί ένα γρήγορο αυτοκίνητο με χαμηλή κατανάλωση, πώς θα αντιμετώπιζε και θα συνδύαζε το σύστημα αυτά τα δύο χαρακτηριστικά;
 - δ) Υλοποιήστε ένα σύστημα γνώσης για επιλογή αυτοκινήτου σε εργαλείο ή γλώσσα της προτίμησής σας, λαμβάνοντας υπόψη τις σχεδιαστικές αποφάσεις που λάβατε στα παραπάνω ερωτήματα. Το σύστημα θα πρέπει να ρωτάει το χρήστη για τις επιλογές του σχετικά με τα επιθυμητά χαρακτηριστικά αλλά και για προσωπικά του στοιχεία που θα καθορίσουν, για παράδειγμα, την προοριζόμενη χρήση του αυτοκινήτου.

2. Να αναπτύξετε ένα σύστημα γνώσης σε εργαλείο ή γλώσσα της προτίμησής σας για την επιλογή ενοικίασης διαμερίσματος. Τα διαθέσιμα διαμερίσματα βρίσκονται καταχωρημένα στη μνήμη γεγονότων. Τα διαθέσιμα στοιχεία για κάθε διαμέρισμα είναι τα εξής:

- Διεύθυνση διαμερίσματος (συμβολοσειρά, παίζει το ρόλο του μοναδικού ονόματος, θεωρήστε ότι υπάρχει ένα διαμέρισμα ανά διεύθυνση)
- Αριθμός υπνοδωματίων (θετικός ακέραιος)
- Όροφος (θετικός ακέραιος ή μηδέν)
- Εμβαδόν (θετικός ακέραιος σε m²)
- Ενοίκιο (ανά μήνα, θετικός ακέραιος σε €)
- Εμβαδόν κήπου (θετικός ακέραιος ή μηδέν σε m²)
- Ύπαρξη ανελκυστήρα (ναι/όχι)
- Βρίσκεται στο κέντρο της πόλης; (ναι/όχι)
- Επιτρέπονται κατοικίδια ζώα στην πολυκατοικία; (ναι/όχι)

Τα διαμερίσματα που θα υπάρχουν στη βάση δεδομένων-γνώσεων του συστήματος γνώσης είναι τα εξής:

Διεύθυνση	Υπνο- δωμάτια	Εμβα- δόν	Κέντρο πόλης	Όρο- φος	Ανελκυ- στήρας	Κατοι- κίδια	Εμβαδόν Κήπου	Ενοίκιο
Βασ.Γεωργίου 35	1	50	ναι	1	όχι	ναι	0	300
Αγγελάκη 7	2	45	ναι	0	όχι	ναι	0	335
Κηφισίας 10	2	65	όχι	2	όχι	ναι	0	350
Πλαστήρα 72	2	55	όχι	1	ναι	όχι	15	330
Τσιμισκή 97	3	55	ναι	0	όχι	ναι	15	350
Πολυτεχνείου 19	2	60	ναι	3	όχι	όχι	0	370
Ερμού 22	3	65	ναι	1	όχι	ναι	12	375

Οι παραπάνω πληροφορίες θα συνδυάζονται με τις απαιτήσεις του επίδοξου ενοικιαστή, οι οποίες θα ερωτώνται από το σύστημα στην αρχή της εκτέλεσης. Οι απαιτήσεις θα είναι οι εξής:

- *Απαιτήσεις σχετικές με το διαμέρισμα*
 - Ελάχιστο επιθυμητό εμβαδόν διαμερίσματος (π.χ. 45m²)
 - Ελάχιστος αριθμός υπνοδωματίων (π.χ. 2)
 - Απαίτηση για ύπαρξη ανελκυστήρα: το σύστημα θα ζητάει από το χρήστη εάν θέλει να υπάρχει ανελκυστήρας στην περίπτωση που το διαμέρισμα βρίσκεται πάνω από κάποιο όροφο (π.χ. αν το διαμέρισμα βρίσκεται από τον 3^ο όροφο και πάνω πρέπει να έχει ανελκυστήρα, ενώ αν βρίσκεται κάτω από τον 3^ο όροφο τότε μπορεί και να μην έχει ανελκυστήρα).
 - Απαίτηση για κατοικίδια ζώα (π.χ. στο διαμέρισμα πρέπει να επιτρέπονται κατοικίδια)
- *Απαιτήσεις σχετικές με το ενοίκιο*
 - Μέγιστο διαθέσιμο ποσό για διαμέρισμα που καλύπτει τις ελάχιστες απαιτήσεις του σχετικά με το εμβαδόν και βρίσκεται στο κέντρο της πόλης (π.χ. αν το διαμέρισμα βρίσκεται στο κέντρο ο πελάτης διατίθεται να δώσει μέχρι 300€)

- Μέγιστο διαθέσιμο ποσό για διαμέρισμα που καλύπτει τις ελάχιστες απαιτήσεις του σχετικά με το εμβαδόν και βρίσκεται στα προάστια (π.χ. αν το διαμέρισμα βρίσκεται στα προάστια ο πελάτης διατίθεται να δώσει μέχρι 250€)
- Μέγιστο επιπλέον διαθέσιμο ποσό για κάθε τετραγωνικό μέτρο εμβαδού διαμερίσματος πέραν της ελάχιστης απαίτησης (π.χ. αν το διαμέρισμα έχει εμβαδόν XXm^2 παραπάνω από την ελάχιστη απαίτηση τότε ο πελάτης διατίθεται να δώσει $XX*5€$ παραπάνω από τα προηγούμενα ποσά)
- Μέγιστο επιπλέον διαθέσιμο ποσό για κάθε τετραγωνικό μέτρο κήπου (π.χ. αν το διαμέρισμα έχει κήπο με εμβαδόν XXm^2 τότε ο πελάτης διατίθεται να δώσει $XX*2€$ παραπάνω από τα προηγούμενα ποσά)
- Άνω όριο διαθέσιμου ποσού (π.χ. μετά τη βασικό ποσό και τις προσαυξήσεις λόγω παραπάνω τετραγωνικών και κήπου, ο πελάτης δεν μπορεί να δώσει πάνω από 400€ το μήνα)

Όλες οι απαιτήσεις να δίνονται από το χρήστη υπό μορφή απαντήσεων σε ερωτήσεις που υποβάλει το σύστημα στην αρχή της εκτέλεσης. Στο τέλος, το σύστημα θα τυπώνει ένα ή περισσότερα διαμερίσματα που ικανοποιούν τις απαιτήσεις του υποψήφιου ενοικιαστή, τυπώνοντάς τα στην οθόνη με ευανάγνωστο τρόπο. Αν δεν υπάρχουν σπίτια που ταιριάζουν στον αγοραστή, τότε να τυπώνεται το ανάλογο μήνυμα, το οποίο θα πρέπει όμως να μην τυπώνεται όταν υπάρχουν κατάλληλα διαμερίσματα.

Αφού τυπωθούν τα διαμερίσματα που ικανοποιούν τις απαιτήσεις του υποψήφιου ενοικιαστή, στη συνέχεια το σύστημα θα πρέπει να τυπώνει στην οθόνη τις διευθύνσεις (μόνο) των διαμερισμάτων που προτείνει προς ενοικίαση βάσει των ακόλουθων προτιμήσεων (οι οποίες υποτίθεται ότι είναι κοινές για όλους τους ενοικιαστές):

- Αν υπάρχει ένα διαμέρισμα φθηνότερο από όλα τα άλλα τότε να προτείνεται αυτό.
- Αν υπάρχουν περισσότερα του ενός φθηνότερα διαμερίσματα, τότε να προτείνεται αυτό που έχει το μεγαλύτερο κήπο.
- Αν υπάρχουν περισσότερα του ενός διαμερίσματα με τα παραπάνω χαρακτηριστικά, τότε να προτείνεται το μεγαλύτερο από αυτά (σε εμβαδόν).

Σε κάθε περίπτωση, μετά την εκτέλεση όλων των παραπάνω να τερματίζεται το πρόγραμμα.

3. Να αναπτυχθεί ένα σύστημα γνώσης σε εργαλείο ή γλώσσα της προτίμησής σας, για διάγνωση βλαβών σε μία οικιακή συσκευή της προτίμησής σας, όπως για παράδειγμα τηλεόραση, πλυντήριο. Η γνώση σχετικά με τα συμπτώματα των βλαβών καθώς και για τις ενέργειες επιδιόρθωσής τους μπορεί να βρεθούν στα εγχειρίδια χρήσης των συσκευών αυτών.
4. Ποια είναι τα πλεονεκτήματα και ποια τα μειονεκτήματα των ακόλουθων γλωσσών προγραμματισμού TN ως γλωσσών ανάπτυξης συστημάτων γνώσης;
 - Λογικός προγραμματισμός.
 - Συναρτησιακός προγραμματισμός.
 - Αντικειμενοστραφής προγραμματισμός.
 - Προγραμματισμός με κανόνες παραγωγής.

5. Ποια προβλήματα μπορεί να προκύψουν σε κάποιο σύστημα γνώσης αν προστεθούν νέοι κανόνες στην υπάρχουσα βάση γνώσης;
6. Αξιολογήστε το εργαλείο τεχνολογίας γνώσης με το οποίο είστε περισσότερο εξοικειωμένοι. Η αξιολόγησή σας πρέπει να περιλαμβάνει:
 - α) Τα χαρακτηριστικά του εργαλείου που σας αρέσουν, εξηγώντας τους λόγους για τους οποίους σας διευκολύνουν στον προγραμματισμό.
 - β) Τα χαρακτηριστικά του εργαλείου που δεν σας αρέσουν, εξηγώντας τους λόγους για τους οποίους σας δυσχεραίνουν το έργο του προγραμματιστή στην επίτευξη συγκεκριμένων εργασιών.
 - γ) Τα χαρακτηριστικά του εργαλείου που δεν υπάρχουν και τα οποία θα θέλατε να έχει. Αν είναι εφικτό διαχωρίστε τα χαρακτηριστικά τα οποία κατά τη γνώμη σας θα μπορούσαν να προστεθούν εύκολα στο εργαλείο και τα χαρακτηριστικά τα οποία θα απαιτούσαν σημαντικό επανασχεδιασμό του εργαλείου ή θα προκαλούσαν απώλεια της αποδοτικότητάς του.
7. Καταγράψτε τις δυσκολίες που αντιμετωπίσατε προκειμένου να μάθετε κάποιο εργαλείο τεχνολογίας γνώσης. Μία πιθανή κατηγοριοποίηση των δυσκολιών που αντιμετωπίζονται θα μπορούσε να είναι η ακόλουθη:
 - *Δυσκολίες περιβάλλοντος.* Δυσκολίες που αντιμετωπίζουν οι χρήστες στο χειρισμό των μενού, του editor, του debugger, κτλ.
 - *Συντακτικές Δυσκολίες.* Δυσκολίες που αντιμετωπίζουν οι χρήστες με τη σύνταξη της γλώσσας αναπαράστασης γνώσης, ιδιαίτερα με την απομνημόνευση των λέξεων κλειδιών, την εισαγωγή των σωστών σημείων στίξης (π.χ. παρενθέσεις, κόμματα, τελείες), κτλ.
 - *Νοηματικές Δυσκολίες.* Δυσκολίες που αντιμετωπίζουν οι χρήστες στην κατανόηση των προγραμματιστικών δυνατοτήτων που προσφέρει το σύστημα και τον τρόπο χειρισμού και αξιοποίησής τους, π.χ. δαίμονες, σύνολα κανόνων, στρατηγικές επίλυσης συγκρούσεων, μέθοδοι εξαγωγής συμπερασμάτων, κτλ.
8. Έστω ότι ένας μηχανικός γνώσης πρόκειται να αναπτύξει ένα σύστημα γνώσης για μία μεγάλη αλυσίδα supermarket. Ένα από τα προβλήματα που αντιμετωπίζει μία τέτοια επιχείρηση είναι η τροφοδοσία των αποθηκών με προϊόντα, η οποία σχετίζεται άμεσα με τον καταλληλότερο τρόπο μεταφοράς των προϊόντων από τον τόπο παραγωγής/προμήθειάς τους στις αποθήκες.

Στο Σχήμα 22.14 φαίνεται ένα πλέγμα ρεπερτορίου, το οποίο συσχετίζει διάφορους τρόπους μεταφοράς προϊόντων, με διάφορες ιδιότητες της μεταφοράς, όπως χρόνος, κόστος, απόσταση, κτλ. Στο ίδιο σχήμα φαίνεται και η βαθμολόγηση που έδωσε ένας ειδικός για τις ιδιότητες των τρόπων μεταφοράς.

Να υπολογιστούν οι αριθμητικές διαφορές μεταξύ των διαφόρων τρόπων μεταφοράς εμπορευμάτων και να συμπληρωθούν στο πλέγμα (Σχήμα 22.15). Στη συνέχεια και βάσει των διαφορών, να βρεθούν ποιοι τρόποι μεταφοράς είναι παρόμοιοι μεταξύ τους και ποιοι είναι τελείως διαφορετικοί.

β)

```
(defrule rule1
  (test1 a)
  (test2 b)
=>
  (assert (suspect yes)) )
(defrule rule2
  (test1 a)
  (not (test2 b))
=>
  (assert (suspect yes)) )
```

γ)

```
(defrule rule1
  (test1 a)
  (test3 c)
  (test2 b)
=>
  (assert (suspect yes)) )
(defrule rule2
  (test2 b)
  (test1 a)
  (test3 c)
=>
  (assert (suspect yes)) )
```

δ)

```
(defrule rule1
  (test1 a)
  (test2 b)
=>
  (assert (suspect yes)) )
(defrule rule2
  (test2 b)
  (test1 a)
=>
  (assert (suspect no)) )
```

ε)

```
(defrule rule1
  (test1 a)
  (test2 b)
=>
  (assert (suspect yes)) )
(defrule rule2
  (test2 b)
  (test1 a)
=>
  (assert (suspect true)) )
```

στ)

```
(defrule rule1
  (test1 ?x a)
  (test2 ?x b)
=>
  (assert (suspect ?x yes)) )
(defrule rule2
  (suspect ?x yes)
  (connected_with ?x ?y)
=>
  (assert (suspect ?y yes)) )
```

ς)

```
(defacts test-facts
  (test1 a)
  (test2 b)
  (test3 c)
  (test4 d) )
(defrule rule1
  (test1 a)
  (test2 b)
=>
  (assert (conclusion1 p)) )
(defrule rule3
  (test4 d)
  (conclusion2 q)
=>
  (assert (conclusion-final w)) )
```

10. Διαπιστώστε εάν υπάρχουν και ποια προβλήματα μοντελοποίησης στους ακόλουθους ορισμούς κλάσεων σε μία οντολογία. Επίσης προτείνετε πιθανούς τρόπους αντιμετώπισης των προβλημάτων.

α)

```
(defclass sensor
  (is-a USER)
  (slot suspect (type SYMBOL) (allowed-symbols yes no) (default no)) )
(defclass internal-component
  (is-a USER)
  (slot suspect (type SYMBOL) (allowed-symbols yes no) (default no))
  (multislot connects_with (type INSTANCE-NAME)) )
```

β)

```
(defclass component
  (is-a USER)
  (slot suspect (type SYMBOL) (allowed-symbols yes no) (default no)) )
(defclass sensor
  (is-a component) )
```

```
(defclass internal-component
  (is-a component)
  (multislot connects_with
    (type INSTANCE-NAME)
    (allowed-classes sensor internal-component) ) )
```

γ)

```
(defclass component
  (is-a USER)
  (slot suspect (type SYMBOL))
  (slot sensed-data (type NUMBER)) )
(defclass sensor
  (is-a component) )
(defclass internal-component
  (is-a component)
  (multislot connects_with (type INSTANCE-NAME)) )
```

11. Στην άσκηση 4 του ΚΕΦΑΛΑΙΟΥ 24 και στην άσκηση 4 του ΚΕΦΑΛΑΙΟΥ 26 περιγράφεται ένα πρόβλημα διάγνωσης χημικών μολύνσεων και εντοπισμού διαρροών από ένα χημικό εργοστάσιο. Να μοντελοποιηθεί σε μία οντολογία η στατική γνώση του προβλήματος, δηλαδή η γνώση για τα χημικά, τις αποθήκες του εργοστασίου, το αποχετευτικό σύστημα και τους σταθμούς μέτρησης στο ποτάμι. Να χρησιμοποιηθεί η μεθοδολογία ανάπτυξης οντολογιών 101 που αναπτύχθηκε στο τρέχον κεφάλαιο και να καταγραφούν αναλυτικά τα βήματά της.
12. Στην άσκηση 5 του ΚΕΦΑΛΑΙΟΥ 26 περιγράφεται ένα πρόβλημα διάγνωσης βλαβών σε ένα ηλεκτρονικό κύκλωμα (Σχήμα 26.2). Να μοντελοποιηθεί σε μία οντολογία η στατική γνώση του προβλήματος, δηλαδή η γνώση για ολόκληρο το ηλεκτρονικό κύκλωμα, τα επί μέρους ολοκληρωμένα κυκλώματα, τους αισθητήρες, τις εισόδους / εξόδους του συστήματος, καθώς και των μετρήσεων σε κάθε κύκλο λειτουργίας του κυκλώματος. Να χρησιμοποιηθεί η μεθοδολογία ανάπτυξης οντολογιών 101 που αναπτύχθηκε στο τρέχον κεφάλαιο και να καταγραφούν αναλυτικά τα βήματά της.