ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ & ΠΛΗΡΟΦΟΡΙΚΗΣ
ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ
2008-2009
1η Σειρά Ασκήσεων

ΑΠΑΝΤΗΣΕΙΣ

1. Μετατρέψτε τις παρακάτω προτάσεις φυσικής γλώσσας (είτε τις Ελληνικές είτε τις Αγγλικές) σε προτάσεις ΚΛΠΤ.

Δίνουμε τις απαντήσεις ακολουθώντας την αναλυτική διαδικασία που αναφέρθηκε στο μάθημα μόνο για την πρόταση 'ε'.

α. "Ο Πλοϋτο αγαπά το αφεντικό του"
"Pluto loves its master"

αγαπά(Πλούτο, αφεντικό_του(Πλούτο))
loves(Pluto, master_of(Pluto))
(Θεωρούμε ότι το «το»/"its" υπονοεί ένα και μοναδικό αφεντικό)

β. "Κάθε σκύλος έχει κάποιο αφεντικό"
"Every dog has a master"

$(\forall x)$ σκύλος(x) $\Rightarrow$ $(\exists y)$ αφεντικό(y, x)
$(\forall x)$ dog(x) $\Rightarrow$ $(\exists y)$ master(y, x)

γ. "Ο Γιάννης ή μισεί τον Γιώργο ή είναι φιλόδοξος"
"John either hates George or is ambitious"

μισεί(Γιάννης. Γιώργος) $\vee$ φιλόδοξος(Γιάννης)
hates(John. George) $\vee$ ambitious(John)

δ. "Τα μήλα είναι ένα είδος τροφής"
"Apples are a kind of food"

τροφή(μήλα)
food(apples)
(Θεωρώντας το «μήλα»/"apples" ως μια οντότητα που αντιπροσωπεύει την έννοια «μήλα»/"apples")
ή
$(\forall x)$ μήλο(x) $\Rightarrow$ τροφή(x)
$(\forall x)$ apple(x) $\Rightarrow$ food(x)
(Θεωρώντας κάθε μήλο ως ξεχωριστή οντότητα)

ε. "Κάθε σαρκοβόρο ζώο τρώει όλα τα ζώα τα μικρότερα από αυτό"
"Every carnivorous animal eats all animals smaller than itself"

Βήμα 1 (Προσδιορισμός κατηγορημάτων/συναρτήσεων)

Κατηγόρημα1: σαρκοβόρο_ζώο
Κατηγόρημα2: ζώο
Κατηγόρημα3: τρώει
Κατηγόρημα4: μικρότερο

Βήμα 2 (Προσδιορισμός αριθμού, τύπου και συμβόλων ορισμάτων συναρτησιακών εκφράεων και κατηγορημάτων)

Κατηγόρημα1: σαρκοβόρο_ζωό
Αριθμός Ορισμάτων: 1
Τύποι Ορισμάτων: μεταβλητή
Σύμβολα Ορισμάτων: x

Κατηγόρημα2: ζώο
Αριθμός Ορισμάτων: 1
Τύποι Ορισμάτων: μεταβλητή
Σύμβολα Ορισμάτων: y

Κατηγόρημα3: τρώει
Αριθμός Ορισμάτων: 2
Τύποι Ορισμάτων: μεταβλητή, μεταβλητή
Σύμβολα Ορισμάτων: x, y

Κατηγόρημα4: μικρότερο
Αριθμός Ορισμάτων: 2
Τύποι Ορισμάτων: μεταβλητή, μεταβλητή
Σύμβολα Ορισμάτων: x, y

Βήμα 3 (Προσδιορισμός ποσοδεικτών μεταβλητών)

x → ∀
y → ∀

Βήμα 4 (Σχηματισμός ατομικών εκφράσεων)

Άτομο1: σαρκοβόρο_ζωό (x)
Άτομο2: ζώο (y)
Άτομο3: τρώει (x, y)
Άτομο4: μικρότερο (y,χ)

Βήμα 5 (Σχηματισμός ομάδων ατόμων ίδιου επιπέδου)

ΟμάδαΑτομ1: { σαρκοβόρο_ζωό (x) ,ζώο (y) , μικρότερο (y,χ)}
ΟμάδαΑτομ2: {τρώει(x, y)}

ή εναλλακτικά

ΟμάδαΑτομ1: { ζώο (y) , μικρότερο (y,χ) ,τρώει(χ,y)}
ΟμάδαΑτομ2: {σαρκοβόρο_ζώο(x)}

Βήμα 6 (Προσδιορισμός συνδετικών ατόμων ομάδων και σχηματισμός τύπων)

ΟμάδαΑτομ1 → Τύπος1-1: (σαρκοβόρο_ζωό (x) $\wedge$ ζώο (y) $\wedge$μικρότερο (y,χ))
ΟμάδαΑτομ2 → Τύπος2-1: τρώει(x, y)

ή εναλλακτικά

ΟμάδαΑτομ1 → Τύπος1-1: {(ζώο (y) $\wedge$ μικρότερο (y,χ)) $\Rightarrow$ τρώει(χ,y)}
ΟμάδαΑτομ2 → Τύπος2-1: {σαρκοβόρο_ζώο(x)}

Βήμα 7 (Προσδιορισμός ομάδων τύπων ίδιου επιπέδου)

ΟμάδαΤυπ1-1: {(σαρκοβόρο_ζωό (x) $\wedge$ ζώο (y) $\wedge$μικρότερο (y,χ)), τρώει(x, y)}

ή εναλλακτικά

ΟμάδαΤυπ1-1: { σαρκοβόρο_ζώο(x), ζώο (y) $\wedge$ μικρότερο (y,χ) $\Rightarrow$ τρώει(χ,y)}

Βήμα 8 (Προσδιορισμός συνδετικών τύπων ομάδας και σχηματισμός τελικού τύπου)

ΟμάδαΤυπ1-1 → Τύπος1-2: (σαρκοβόρο_ζωό (x) $\wedge$ ζώο (y) $\wedge$μικρότερο (y,χ)) $\Rightarrow$ τρώει(x, y)

ή εναλλακτικά

ΟμάδαΤυπ1-1 → Τύπος1-2: σαρκοβόρο_ζώο(x) $\Rightarrow$(ζώο (y) $\wedge$ μικρότερο (y,χ)) $\Rightarrow$ τρώει(χ,y)}

Βήμα 10 (Σχηματισμός τελικής πρότασης)

($\forall$x) ($\forall$y) ((σαρκοβόρο_ζωό (x) $\wedge$ ζώο (y) $\wedge$μικρότερο (y,χ)) $\Rightarrow$ τρώει(x, y))

ή εναλλακτικά

($\forall$x) (σαρκοβόρο_ζώο(x) $\Rightarrow$(($\forall$y) (ζώο (y) $\wedge$ μικρότερο (y,χ))$\Rightarrow$ τρώει(χ,y)))

στ. "Κανείς άνδρας δεν συμπαθεί μια γυναίκα που είναι χορτοφάγος"
"No man likes a woman who is vegetarian"

(∀x)(∀y)((άνδρας(x) ∧ γυναίκα(y) ∧ χορτοφάγος(y)) ⇒ ¬συμπαθεί(x,y))
(∀x)(∀y)((man(x) ∧ woman(y) ∧ vegeterian(y)) ⇒ ¬likes(x,y))
(Υπάρχουν και άλλες ισοδύναμες εκφράσεις-βλ. περυσινές απαντήσεις)

2. Να μετατραπούν σε προτασιακή μορφή οι παρακάτω προτάσεις ΚΛΠΤ.

α. (∀x) (∀y) (∀z) ((pet(x) ∧ master(x, y) ∧ lives(y, z)) ⇒ lives(x, z))

1. Απαλοιφή συνεπαγωγών:
(∀x) (∀y) (∀z) (¬(pet(x) ∧ master(x,y) ∧lives(y,z)) ∨ lives(x,z))

2. Περιορισμός εμβέλειας του ~:
(∀x) (∀y) (∀z) ((¬pet(x) ∨ ¬master(x,y) ∨ ¬lives(y,z)) ∨ lives(x,z))

3. Μετονομασία μεταβλητών:
Μη εφαρμόσιμο

4. Μετατροπή σε ΚΜΡ
Μη εφαρμόσιμο

5. Απαλοιφή υπαρξιακών ποσοσδεικτών
Μη εφαρμόσιμο

6. Απαλοιφή καθολικών πσοσδεικτών
((¬pet(x) ∨ ¬master(x,y) (¬lives(y,z)) (lives(x,z))

7. Μετατροπή σε ΚΣΜ
Μη εφαρμόσιμο

8. Απαλοιφή διασυνδεκτικών και καταγραφή προτάσεων
φ ≡ (¬pet(x), ¬master(x,y), ¬lives(y,z), lives(x,z))

9. Μετονομασία μεταβλητών
Μη εφαρμόσιμο (μία μόνο πρόταση)

β. ((∀x) ((∃y) a(y) ⇒ b(x, y))) ∨ ((∀x) c(x))

1. Απαλοιφή συνεπαγωγών:
((∀x) (((y) (¬a(y) (b(x, y)))) ((((x) c(x))

2. Περιορισμός εμβέλειας:
Μη εφαρμόσιμο

3. Μετονομασία μεταβλητών:
(((x) (((y) (¬a(y) (b(x, y)))) ((((z) c(z))

4. Μετατροπή σε ΚΜΡ
((x)((y)((z) (¬a(y) (b(x, y) (c(z))

5. Απαλοιφή υπαρξιακών ποσοδεικτών {y = f(x)}
(∀x)(∀z) (¬a(f(x)) ∨ b(x,f(x)) ∨ c(z))

6. Απαλοιφή καθολικών πσοσδεικτών
(¬a(f(x)) ∨ b(x,f(x)) (c(z))

7. Μετατροπή σε ΚΣΜ
Μη εφαρμόσιμο

8. Απαλοιφή διασυνδεκτικών και καταγραφή προτάσεων
φ ≡ (¬a(f(x)) , b(x, f(x)) , c(z))

9. Μετονομασία μεταβλητών
Μη εφαρμόσιμο (μία μόνο πρόταση)

3. Προσπαθήστε να ενοποιήσετε τα παρακάτω ζεύγη στοιχείων. Αν ενοποιούνται, βρείτε την γενικότερη ενοποιήτρια. Αν όχι, εξηγείστε γιατί.

α. p(x, y) , p(a, z)
Ενοποιούνται με γενικότερη ενοποιήτρια την σ = {a/x, y/z}

β. p(x, x) , p(a, b)
Δεν ενοποιούνται διότι δεν υπάρχει γ.ε. Αν υπήρχε, θα ήταν η «σ = {a/x, b/x}», η οποία όμως δεν είναι έγκυρη, διότι η μεταβλητή «x» έχει σαν προσδέσεις δύο διαφοερετικές σταθερές, επομένως δημιουργείται σύγκρουση.

γ. απόγονος(x, πατέρας-του(x)) , απόγονος(γιάννης, βασίλης)
Δεν ενοποιούνται., διότι ενώ η «x» ενοποιείται με την σταθερά «γιάννης», η συνάρτηση «πατέρας-του(x)» δεν μπορεί να ενοποιηθεί με μια σταθερά (την «βασίλης»).

δ. απόγονος(x, y) , απόγονος(βασίλης, πατέρας-του(βασίλης))
Ενοποιούνται με γ.ε. την σ = {βασίλης/x, πατέρας-του(βασίλης)/y}

ε. q(x, a, y) , q(z, z, b)
Και αυτά ενοποιούνται με γ.ε. την σ = {z/x, a/z, b/y}, η οποία καταλήγει στην σ = {a/x, a/z, b/y}

στ. q(x) , ¬q(a)
Δεν ενοποιούνται, επειδή έχουν αντίθετη πολικότητα.

4. Δίνονται οι παρακάτω προτάσεις ΚΛΠΤ

(1) works-in(george, patras)

(2) works-in(paul, rio)

(3) master(george, pluto)

(4) master(paul, boby)

(5) ((x)((y)(works-in(x, y) (lives-in(x, y))

(6) ((x)((y)((z)((master(x, y) (lives-in(x, z)) (lives-in(y, z))

όπου x, y, z είναι μεταβλητές.

(α) Χρησιμοποιώντας τη μέθοδο της αντίφασης της επίλυσης, αποδείξτε ότι “Pluto lives in Patra”

Πρώτα πρέπει να τις μετατρέψουμε σε προτασιακή μορφή:

(1) works-in(george, patras)

(2) works-in(paul, rio)

(3) master(george, pluto)

(4) master(paul, boby)

(5) (¬works-in(x1, y1), lives-in(x1, y1))

(6) (¬master(x2, y2), ¬lives-in(x2, z), lives-in(y2, z))

Η πρόταση που θέλουμε να αποδείξουμε είναι η φ = lives-in(pluto, patras) και το σύνολο των αξιωμάτων S με βάση τα οποία θα την αποδείξουμε είναι οι προτάσεις του (α). Παίρνουμε την άρνηση της φ, ¬φ = ¬lives-in(pluto, patras) και, αφού βρίσκεται ήδη σε προτασιακή μορφή, την προσθέτουμε στο S. Στη συνέχεια, εφαρμόζοντας διαρκώς επίλυση μεταξύ προτάσεων του S προσπαθούμε να καταλήξουμε στην κενή πρόταση (). Κάθε επιλύουσα που δεν είναι η κενή πρόταση προστίθεται στο S και η διαδικασία συνεχίζεται. Αν προκύψει επιλύουσα () σταματάμε την διαδικασία των επιλύσεων αφού η εξαγωγή της κενής πρότασης από το S∪¬φ σημαίνει πως η φ συνεπάγεται λογικά από το S.

Έχουμε:

(1) (works-in(george, patras))

(2) (works-in(paul, rio))

(3) (master(george, pluto))

(4) (master(paul, boby))

(5) (¬works-in(x1, y1) , lives-in(x1, y1))

(6) (¬master(x2, y2) , ¬lives-in(x2, z), lives-in(y2, z))

(7) (¬lives-in(pluto, patras))

---

(8) (¬lives-in(george, z), lives-in(pluto, z)) (3, 6) με σ={george/x2 , pluto/y2}

(9) (lives-in(george, patras)) (1, 5) με σ={george/x1 , patras/y1}

(10) (lives-in(pluto, patras)) (8, 9) με σ={patras/z}

(11) ( ) (7, 10)

Οπότε αποδεικνύεται η πρόταση φ.

Αφού εδώ δεν μας περιορίζει κάποια στρατηγική στα βήματα που θα κάνουμε, οποιαδήποτε σειρά επιλύσεων διαλέξει κανείς είναι σωστή, αρκεί να οδηγεί στην () με σωστές επιλύσεις.

(β) Χρησιμοποιώντας τη μέθοδο της αντίφασης της επίλυσης, απαντήστε στο ερώτημα "Who lives in Rio?" (βρείτε δηλ. τις τιμές της μεταβλητής x για τις οποίες αληθεύει η πρόταση (∃x) lives-in(x, rio)).

Ακολουθούμε την προηγούμενη διαδικασία με την διαφορά πως εδώ παίζουν μεγαλύτερο ρόλο από πριν οι αντικαταστάσεις που έχουν γίνει όταν φτάνουμε στην κενή πρόταση. Από αυτές βρίσκουμε τις ζητούμενες τιμές του x. Μια ακόμα διαφορά είναι πως τώρα δεν θα σταματήσουμε την πρώτη φορά που θα καταλήξουμε στην (). Θα επαναλάβουμε την διαδικασία για να δούμε με πόσες διαφορετικές τιμές του x οδηγούμαστε σε απόδειξη της πρότασης.

Η πρόταση που θέλουμε να αποδείξουμε είναι η φ = (∃x) lives-in(x, rio).
Παίρνουμε την άρνηση της φ, ¬φ = ¬(∃x) lives-in(x, rio) = (∀x) ¬lives-in(x, rio), την μετατρέπουμε σε προτασιακή μορφή : ¬lives-in(x, rio) και την προσθέτουμε στο S.

Έχουμε:

(1) (works-in(george, patras))
(2) (works-in(paul, rio))
(3) (master(george, pluto))
(4) (master(paul, boby))
(5) (¬works-in(x1, y1) , lives-in(x1, y1))
(6) (¬master(x2, y2) , ¬lives-in(x2, z), lives-in(y2, z))
(7) (¬lives-in(x, rio))

---

(8) (lives-in(paul, rio)) (2, 5) με σ={paul/x1 , rio/y1}
(9) ( ) (7, 8) με σ={paul/x}
Η διαδικασία επαναλαμβάνεται για την επόμενη τιμή του x:

(8) (¬lives-in(paul, z), lives-in(boby, z)) (4, 6) με σ={paul/x2 , boby/y2}
(9) (¬works-in(paul, z), lives-in(boby, z)) (5, 8) με σ={paul/x1 , z/y1}
(10) (lives-in(boby, rio)) (2, 9) με σ={rio/z}
(11) ( ) (7, 8) με σ={boby/x}

Εδώ μπορούμε να σταματήσουμε αφού συνεχίζοντας θα βρούμε ξανά τις ίδιες τιμές:

(8) (¬works-in(x, rio)) (5, 7) με σ={x/x1 , rio/y1}
(9) ( ) (2, 8) με σ={paul/x}

και

(8) (¬master(x2, x), ¬lives-in(x2, rio)) (6, 7) με σ={x/y2 , rio/z}
(9) (¬works-in(x2, rio), ¬master(x2, x)) (5, 8) με σ={x2/x1 , rio/y1}
(10) (¬master(paul, x)) (2, 9) με σ={paul/x2}
(11) ( ) (4, 10) με σ={boby/x}

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ & ΠΛΗΡΟΦΟΡΙΚΗΣ
ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ
2008-2009
2η Σειρά Ασκήσεων

ΑΠΑΝΤΗΣΕΙΣ

1. Δίνεται το παρακάτω σύνολο προτάσεων :
$T \Rightarrow (R \vee Q)$, $(P \wedge \neg Q) \Rightarrow R$, $(\neg T \wedge \neg P) \Rightarrow \neg S$, $\neg Q$, S, $\neg T$
(α) Να μετατραπούν σε προτασιακή μορφή.
(β) Ποια από τις στρατηγικές N1–Επίλυση και Μοναδιαία Επίλυση είναι ασφαλέστερη για αυτό το σύνολο προτάσεων και γιατί;
(γ) Χρησιμοποιείστε τη διαδικασία της αντίφασης της επίλυσης και την ασφαλέστερη στρατηγική που επιλέξατε στο (β) για να αποδείξετε ότι η R είναι θεώρημα.

α) Το σύνολο σε προτασιακή μορφή είναι:
{¬T ,R ,Q},{ ¬P ,Q ,R, },{T ,P ¬S,}{ ¬Q},{ S},{ ¬T}

β)Η N1-Επίλυση είναι πάντα πλήρης. Αντίθετα, η Μοναδιαία είναι πλήρης μόνο για προτάσεις τύπου Horn. Όμως, για παράδειγμα, η πρόταση {¬T ,R ,Q} δεν είναι τύπου Horn αφού έχει περισσότερα από ένα θετικά στοιχεία, επομένως η Μοναδιαία επίλυση δεν είναι πλήρης για το συγκεκριμένο σύνολο προτάσεων. Άρα ασφαλέστερη στρατηγική είναι η N1-Επίλυση που είναι πλήρης.

γ) Σύμφωνα με την διαδικασία της αντίφασης της επίλυσης, τοποθετούμε την άρνηση της προς απόδειξη πρότασης στο αρχικό σύνολο προτάσεων και επιλύοντας σύμφωνα με την επιλεγείσα στρατηγική προσπαθούμε να καταλήξουμε στην κενή πρόταση.

Να θυμηθούμε στο σημείο αυτό ότι στην N1-Επίλυση πρέπει σε κάθε βήμα της επίλυσης τουλάχιστον ένας απ' τους γονείς να είναι αρνητική πρόταση, δηλαδή να είναι μια πρόταση με όλα της τα στοιχεία αρνητικά.

Να επισημάνουμε επίσης ότι πρέπει να επιλύουμε οτιδήποτε μπορεί να επιλυθεί ακόμα και αν μας οδηγεί σε πρόταση με περισσότερα στοιχεία απ' τους γονείς της. Αν δεν το κάνουμε αυτό μπορεί να χάσουμε λύσεις ή ίσως στην χειρότερη περίπτωση να μην ακολουθήσουμε ποτέ την μοναδική διαδρομή που θα μας οδηγούσε σε λύση.

1. (¬T, R, Q)
2. (¬P, Q, R)
3. (T, P ¬S)
4. (¬Q)
5. (S)
6. (¬T)
7. (¬R)

---

8. (¬T, R) (1,4)
9. (¬T, Q) (1,7)
10. (¬P, R) (2,4)
11. (¬P, Q) (2,7)
12. (P, ¬S) (3,6)

---

13. (¬T) (9,7) ΑΡΝΗΤΙΚΗ ΠΡΟΤΑΣΗ
14. (¬P) (10,7) ΑΡΝΗΤΙΚΗ ΠΡΟΤΑΣΗ

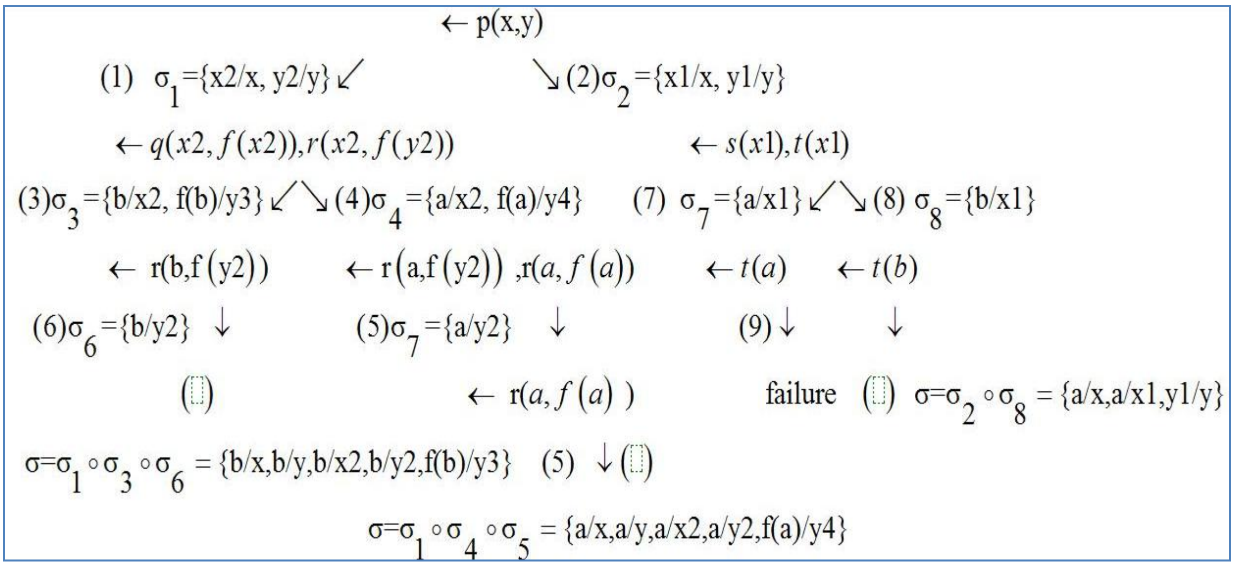
15. (¬S) (14,12) ΑΡΝΗΤΙΚΗ ΠΡΟΤΑΣΗ
16. () (15,5)

Επομένως, η R είναι θεώρημα.

2. Δίνεται το παρακάτω λογικό πρόγραμμα:

(1) p(x2, y2) (q(x2, f(x2)), r(x2, f(y2))
(2) p(x1, y1) (s(x1), t(x1)
(3) q(b, y3)
(4) q(a, y4) (r(a, f(a))
(5) r(a, f(a))
(6) r(b, f(b))
(7) s(a)
(8) s(b)
(9) t(b).
(α) Αν ο στόχος είναι (p(x, y), σχεδιάστε το SLD-δέντρο.
(β) Με ποιά στρατηγική από τις depth-first with backtracking και breadth-first βρίσκουμε γρηγορότερα (δηλ. σε λιγότερα βήματα) τη λύση; Εξηγείστε.
(γ) Τι μπορούμε να κάνουμε για μειώσουμε τα βήματα και στις δύο περιπτώσεις;

(α) Το SLD-δέντρο



Συνεπώς, η πρόταση p(x,y) αληθεύει αν το x έχει την τιμή a και το y έχει οποιαδήποτε τιμή, ή αν x = y = b.

β) Με ποιά στρατηγική από τις depth-first with backtracking και breadth-first βρίσκουμε γρηγορότερα (δηλ. σε λιγότερα βήματα) τη λύση; Εξηγείστε.

Χρησιμοποιώντας depth-first with backtracing (κατά βάθος με οπισθοδρόμηση) βρίσκουμε την πρώτη λύση σε 3 βήματα - επιλύσεις, τις (1) - (3) - (6).
Χρησιμοποιώντας breadth-first βρίσκουμε την ίδια λύση σε 7 βήματα – επιλύσεις, τις (1) - (2) - (3) - (4) - (7) - (8) - (6) .
Συνεπώς, με τη πρώτη στρατηγική βρίσκουμε γρηγορότερα τη λύση.

(γ) Τι μπορούμε να κάνουμε για μειώσουμε τα βήματα και στις δύο περιπτώσεις;

Συνήθως, για να μειώσουμε τα βήματα εύρεσης της λύσης το μόνο που μπορούμε να κάνουμε είναι να αλλάξουμε την σειρά κάποιων προτάσεων ή την σειρά κάποιων όρων στις προτάσεις. Στην συγκεκριμένη περίπτωση, όσες εναλλαγές και να κάνουμε θα παρατηρήσουμε πως δεν μειώνονται τα βήματα καμίας από τις παραπάνω στρατηγικές.

3. Δίνεται η παρακάτω ιστορία: «Κάποιος (Ε) παντρεύτηκε μια χήρα (Η) που είχε μια μεγάλη κόρη (Κ). Ο πατέρας του Ε (Π) συμπάθησε και παντρεύτηκε την θετή κόρη (Κ) του Ε. Η σύζυγος του Ε (Η) γέννησε ένα γυιό (Γ).»
(α) Σχηματίστε προτάσεις Prolog που να εκφράζουν τα γεγονότα που πηγάζουν από την παραπάνω ιστορία.
(β) Εκφράστε σε κανόνες Prolog τις απαραίτητες οικογενειακές σχέσεις που πρέπει να συμπληρώσουν τα παραπάνω γεγονότα.
(γ) Σχηματίστε το πρόγραμμα Prolog που προκύπτει από τα (α) και (β). Υλοποιείστε σε Prolog το ερώτημα «Ποιός είναι θείος του Ε;» και τρέξτε το πρόγραμμα. Ποια είναι η απάντηση; Εξηγείστε τα βήματα της Prolog.

a)-b)

```
% Author:
male(e).
male(p).
male(g).
female(h).
female(k).
married(e,h).
married(p,k).
parent(h,k).
parent(p,e).
parent(e,g).
parent(h,g).

stepparent(X,Y):-parent(Z,Y), married(X,Z), \+parent(X,Y);
parent(Z,Y), married(Z,X), \+parent(X,Y).
```

```
father(X,Y):-parent(X,Y), male(X).
mother(X,Y):-parent(X,Y), female(X).
son(X,Y):-parent(Y,X), male(X).
daughter(X,Y):-parent(Y,X), female(X).
grandfather(X,Y):-parent(X,Z), parent(Z,Y), male(X).
grandmother(X,Y):-parent(X,Z), parent(Z,Y), female(X).
brother(X,Y):-parent(Z,Y), parent(Z,X), \+X==Y, male(X).
sister(X,Y):-parent(Z,Y), parent(Z,X), \+X==Y, female(X).
wife(X,Y):-married(Y,X), male(Y), female(X).
husband(X,Y):-married(X,Y), male(X),female(Y).
uncle(X,Y):-parent(Z,Y), brother(X,Z); stepparent(Z,Y), brother(X,Z).
aunt(X,Y):-parent(Z,Y), sister(X,Z); stepparent(Z,Y), sister(X,Z).
```

- ?-uncle(X, e).
- Απάντηση prolog
- X=g;
  false
- ο θείος του Ε είναι ο Γ

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ & ΠΛΗΡΟΦΟΡΙΚΗΣ
ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ
2008-2009
3η Σειρά Ασκήσεων

ΑΠΑΝΤΗΣΕΙΣ

1. Δίνεται το προβλήματος των τριών δοχείων γάλακτος: «Υπάρχουν τρία δοχεία χωρητικότητας 8, 5 και 2 λίτρων αντίστοιχα που δεν διαθέτουν μετρητή. Το δοχείο των 8 λίτρων είναι γεμάτο με γάλα. Θέλουμε να μοιράσουμε το γάλα σε δύο ίσες ποσότητες στα δύο μεγαλύτερα δοχεία. Οι δυνατές ενέργειες αφορούν άδειασμα ποσότητας γάλακτος από το ένα δοχείο σε άλλο.»

Ζητούνται:

α) Να ορίσετε (α1) ένα τρόπο-δομή αναπαράστασης μιας (τυχαίας) κατάστασης (α2) την αρχική κατάσταση και την/τις τελική/ές κατάσταση/εις και (α3) τους τελεστές μετάβασης.

Ορίζουμε σαν αναπαράσταση μιας τυχαίας κατάστασης την: (x, y,z) όπου x η ποσότητα γάλα στο δοχείο των 8 λίτρων, y η ποσότητα γάλα στο δοχείο των 5 λίτρων και z η ποσότητα στο δοχείο των 2 λίτρων.

α1) Αρχική κατάσταση: (8,0,0)

α2) Τελικές καταστάσεις : (4,4,0)

α3) Τελεστές Μετάβασης:

ΤΕΛΕΣΤΕΣ-ΠΕΡΙΓΡΑΦΗ	ΠΡΟΫΠΟΘΕΣΕΙΣ	ΑΠΟΤΕΛΕΣΜΑ
Τ1: άδειασε το δοχείο Α στο Β	x>0,y<5	Αν x<5-y τότε (0,y+x,z) αλλιώς (x-(5-y),5,z)
Τ2: άδειασε το δοχείο Α στο Γ	x>0,z<2	Αν x<2-z τότε(0,y,x+z) αλλιώς (x-(2-z),y,2)
Τ3: άδειασε το δοχείο Β στο Α	y>0,x<8	Αν y<8-x τότε (x+y,0,z) αλλιώς (8,y-(8-x),z)
Τ4: άδειασε το δοχείο Β στο Γ	y>0,z<2	Αν y<2-z τότε (x,0,y+z) αλλιώς (x,y-(2-z),2)
Τ5: άδειασε το δοχείο Γ στο Α	z>0,x<8	Αν z<8-x τότε (x+z,y,0) αλλιώς (8,y,z-(8-x))
Τ6: άδειασε το δοχείο Γ στο Β	z>0,y<5	Αν z<5-y τότε (x,y+z,0) αλλιώς (x,5,z-(5-y))

β) Προσδιορίστε τον χώρο καταστάσεων του προβλήματος. Υπάρχουν ανέφικτες καταστάσεις; Αν ναι, μπορούν να προσδιοριστούν;

Χώρος Καταστάσεων του προβλήματος είναι (x,y,z): $x \in \{0,1,2,3..8\}$, $y \in \{0,1,2,3,4,5\}$, $z \in \{0,1,2\}$ με $x+y+z=8$
Ανέφικτες καταστάσεις είναι αυτές στις οποίες, ξεκινώντας από οποιαδήποτε αρχική κατάσταση και εφαρμόζοντας τους τελεστές μετάβασης, δεν θα φτάσουμε ποτέ. Για να τις βρούμε πρέπει να αναπτύξουμε τον γράφο.

γ) Ορίστε μια συνάρτηση κόστους g(n) και μια ευρετική συνάρτηση h(n).
Συνάρτηση κόστους:
Η συνάρτηση κόστους σχετίζεται με το **κόστος μετάβασης**. Το κόστος μετάβασης

Η συνάρτηση κόστους σχετίζεται με το κόστος μετάβασης που χαρακτηρίζει μια μετάβαση από μία κατάσταση n'=(x',y',z') σε άλλη n=(x,y,z).
Στην περίπτωσή μας το κόστος σχετίζεται με τον όγκο του γάλακτος που μεταβιβάζεται.

$g(n',n) = (|x'-x|+|y'-y|+|z'-z|)/2$ (επειδή σε κάθε μετάβαση συμμετέχουν 2 δοχεία)

Επομένως το συνολικό κόστος μετάβασης (δηλ. το κόστος μετάβασης από την αρχή) στην κατάσταση n είναι: **$g(n)=g(n')+g(n',n)$.**

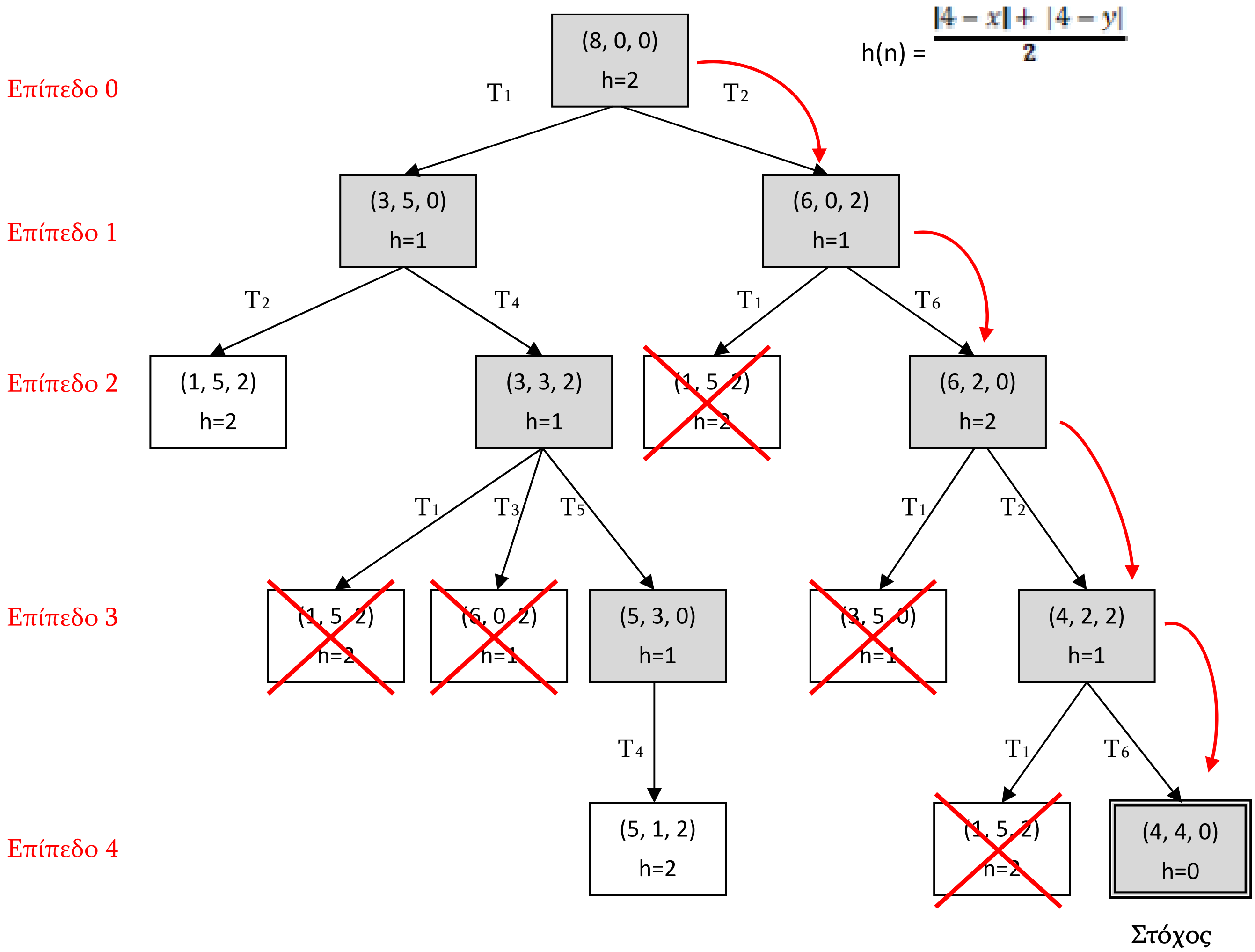
Σαν ευρετικό, σε μια κατάσταση (x, y,z), θεωρούμε μια συνάρτηση που μας εκτιμά το πόσο «απέχει» από την τελική κατάσταση μια τυχαία κατάσταση. Θέλουμε δηλ. να είναι h(n)=0 στις καταστάσεις-στόχους.
Μια ευρετική συνάρτηση είναι η εξής:

$h(n)= (|4-x|+|4-y|)/2$

δ) Εφαρμόστε τα τρία πρώτα βήματα των αλγορίθμων (δ1) αναζήτηση δέσμης και (δ2) διακλάδωση και δέσμευση, σχεδιάζοντας τα αντίστοιχα επίπεδα στο δέντρο καταστάσεων.
Αναζήτηση Δέσμης

Ο αλγόριθμος αναζήτησης δέσμης επιλέγει τα m καλυτέρα παιδιά (με βάση το h(n)) από κάθε επίπεδο για περαιτέρω ανάπτυξη. Τα υπόλοιπα τα διαγράφει. Επίσης, καθορίζουμε τα εξής για αποδοτικότερη λειτουργία του αλγορίθμου:
(α) Διαγράφονται όσες καταστάσεις έχουν ήδη αναπτυχθεί σε προηγούμενο επίπεδο είτε είναι στο ίδιο μονοπάτι είτε όχι (αφού μας ενδιαφέρει να βρούμε μια λύση).
(β) Διαγράφεται κάθε ίδια κατάσταση που παράγεται στο ίδιο επίπεδο, δηλ. κρατάμε μόνο την πρώτη που παράγεται.
(γ) Στην περίπτωση δύο ή περισσότερων καταστάσεων που έχουν την ίδια ευρετική τιμή προτιμούμε τέτοια διάταξη, ώστε μια κατάσταση να ανήκει σε διαφορετικό γονέα από την προηγούμενη και την επόμενή της.
(δ) Μεταξύ δύο καταστάσεων του ίδιου γονέα με την ίδια ευρετική συνάρτηση επιλέγουμε την πρώτη αριστερά.
Στη συνέχεια, φαίνεται η ανάπτυξη (όλου) του δέντρου για m=2, όπου δεν αναγράφονται καθόλου τα κλαδιά που οδηγούν σε καταστάσεις που έχουν ήδη αναπτυχθεί (δηλ. όσες

εμπίπτουν στην περίπτωση (α)), με κόκκινο Χ παριστάνεται η διαγραφή καταστάσεων της περίπτωσης (β) και με γκρι εμφανίζονται οι καταστάσεις που επιλέγονται σε κάθε επίπεδο.



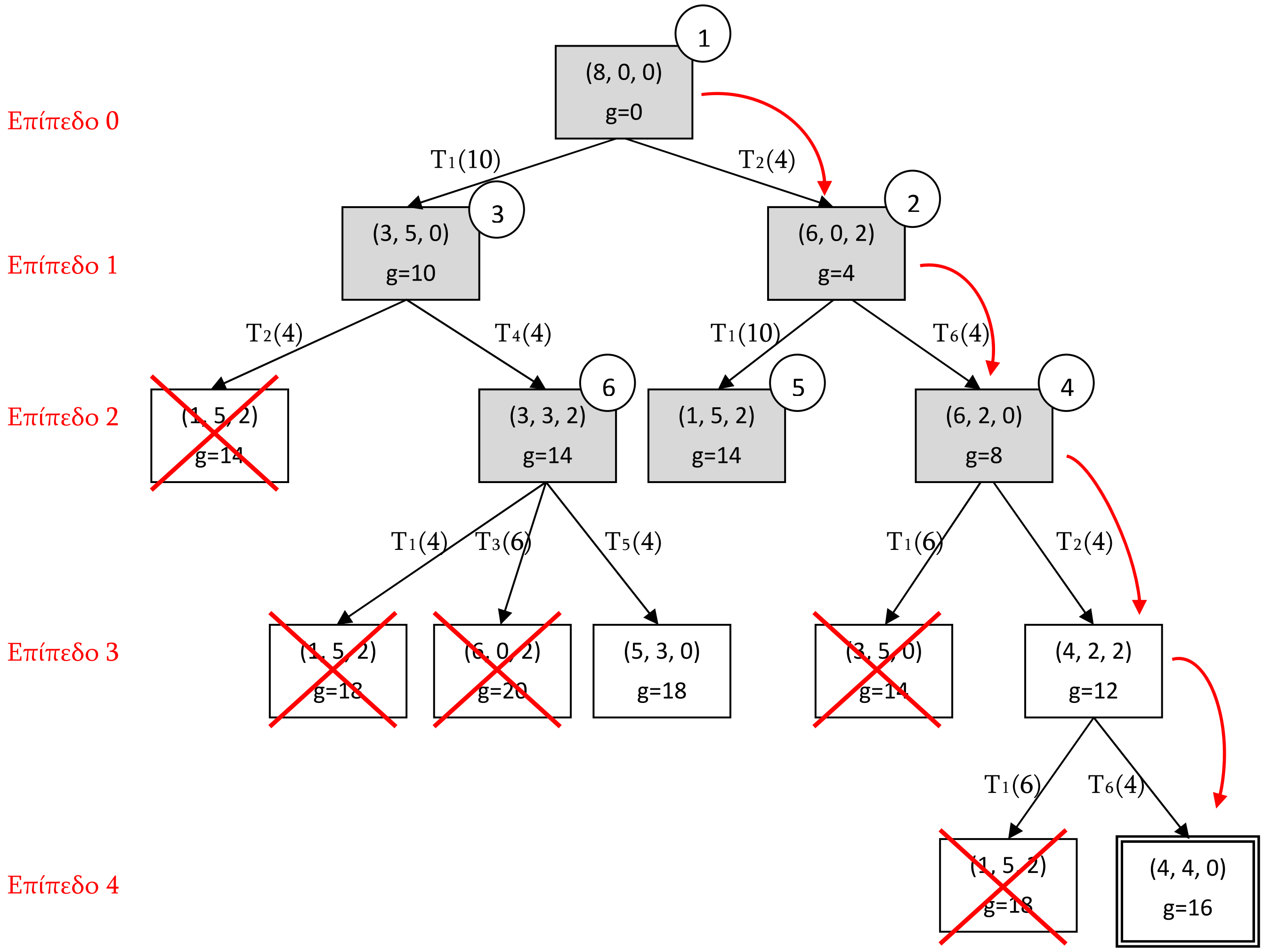
Η διαδρομή που βρέθηκε είναι η T_2 - T_6 - T_2 - T_6. Δηλαδή,

Άδειασε το Α στο Β
Άδειασε το Γ στο Β
Άδειασε το Α στο Β
Άδειασε το Γ στο Β

Αλγόριθμος Διακλάδωσης και Δέσμευσης (ή Επέκτασης και Οριοθέτησης)

Ο αλγόριθμος διακλάδωσης και δέσμευσης επιλέγει κάθε φορά από τις τρέχουσες ανοικτές καταστάσεις (που δεν έχουν αναπτυχθεί) την κατάσταση που «απέχει» λιγότερο από την αρχή, με βάση τη συνάρτηση κόστους g(n). Ως γνωστόν, βρίσκει τη βέλτιστη λύση (δηλ. τη λύση με το μικρότερο κόστος), αν η συνάρτηση κόστους είναι σωστή. Θεωρούμε παρόμοια με τον προηγούμενο αλγόριθμο:
(α) Διαγράφονται όσες καταστάσεις έχουν ήδη αναπτυχθεί σε προηγούμενο επίπεδο.
(β) Διαγράφεται κάθε ίδια ανοικτή κατάσταση με μεγαλύτερο κόστος.

(γ) Στην περίπτωση δύο ή περισσότερων ίδιων καταστάσεων που έχουν το ίδιο κόστος διαγράφουμε αυτή που παρήχθη πιο πρόσφατα.
(δ) Στην περίπτωση δύο ή περισσότερων καταστάσεων που έχουν το ίδιο κόστος προτιμούμε αυτή που παρήχθη παλαιότερα.



Η διαδρομή που βρέθηκε, και είναι η βέλτιστη, φαίνεται με την κόκκινη γρα Στόχος
Παρατηρείστε ότι ο αλγόριθμος αναζήτησης δέσμης βρήκε κατά σύμπτωση τη βέλτ
λύση:
T_2 - T_6 - T_2 - T_6

2. Δίνεται το γνωστό πρόβλημα του τετράγωνου παζλ: «Έχουμε ένα τετράγωνο πλαίσιο 3Χ3=9 τετράγωνων θέσεων, που τις 8 θέσεις καταλαμβάνουν τετράγωνα πλακίδια αριθμημένα από το 1 ως το 8, ενώ η ένατη είναι κενή. Τα πλακίδια μπορούν να μετακινούνται πάνω, κάτω, δεξιά και αριστερά, εφ' όσον αυτό είναι δυνατόν».

2	8	3
1	6	4
7		5

(α)

→

1	2	3
8		4
7	6	5

(β)

Ζητούνται:

α) Να περιγραφεί σαν πρόβλημα αναζήτησης, δηλ. να οριστούν η αρχική κατάσταση, η τελική κατάσταση, οι τελεστές μετάβασης, μια συνάρτηση κόστους g(n) και μια ευρετική συνάρτηση h(n).

(α) Η <u>αναπαράσταση</u> μιας τυχαίας κατάστασης του προβλήματος μπορεί να είναι η εξής:

(Α1, Α2, Α3, Β1, Β2, Β3, Γ1, Γ2, Γ3)

όπου τα γράμματα αντιστοιχούν στις γραμμές και οι αριθμοί στις στήλες, ώστε να οριστεί κάθε κελί του πίνακα. Πχ. Α1 → πρώτο κελί ($1^η$ γραμμή - $1^η$ στήλη):

Α1	Α2	Α3
Β1	Β2	Β3
Γ1	Γ2	Γ3

Οπότε, <u>αρχική κατάσταση</u>: (2, 8, 3, 1, 6, 4, 7, _, 5), <u>τελική κατάσταση</u>: (1, 2, 3, 8, _, 4, 7, 6, 5)
Οι τελεστές μετάβασης είναι οι παρακάτω.

Α/Α	ΠΕΡΙΓΡΑΦΗ	ΠΡΟΫΠΟΘΕΣΕΙΣ	ΑΠΟΤΕΛΕΣΜΑ
Τ1	Μετακίνηση του κενού προς τα πάνω	Βi = “ ” ή Γi = “ ” για i=1,2,3	(Α1, Α2, Α3, Β1, Β2, Β3, Γ1, Γ2, Γ3)
Τ2	Μετακίνηση του κενού προς τα κάτω	Αi = “ ” ή Βi = “ ” για i=1,2,3	(Α1, Α2, Α3, Β1, Β2, Β3, Γ1, Γ2, Γ3)
Τ3	Μετακίνηση του κενού προς τα δεξιά	Αi = “ ” ή Βi = “ ” ή Γi = “ ” για i=1,2	(Α1, Α2, Α3, Β1, Β2, Β3, Γ1, Γ2, Γ3)
Τ4	Μετακίνηση του κενού προς τα αριστερά	Αi = “ ” ή Βi = “ ” ή Γi = “ ” για i=2,3	(Α1, Α2, Α3, Β1, Β2, Β3, Γ1, Γ2, Γ3)

Συνάρτηση Κόστους
Δεν υπάρχει κάτι που να διαφοροποιεί το «κόστος» (π.χ. τον κόπο ή το έργο που απαιτείται) στις διάφορες μεταβάσεις, οπότε εδώ θεωρούμε ότι το κόστος μετάβασης είναι πάντα 1, δηλ. g(n) = 1 + g(n-1) για κάθε κατάσταση n (όπου n-1 είναι η κατάσταση-γονέας της n).

Ευρετική Συνάρτηση
Σαν ευρετική συνάρτηση, δηλ. μια συνάρτηση που μετρά την απόσταση κάθε κατάστασης από τον στόχο (δηλ. την τελική κατάσταση), ορίζουμε τη συνάρτηση που να μετρά τον αριθμό τον κελιών της τρέχουσας κατάστασης έχουν ίδιο περιεχόμενο με τα αντίστοιχα κελιά της τελικής κατάστασης. Έτσι, αν x(n) ο αριθμός τον κελιών που συμφωνούν με την κατάσταση-στόχο, η ευρετική συνάρτηση είναι η εξής:

$$h(n) = 9 - x(n)$$

Η ευρετική αυτή συνάρτηση επιστρέφει για κάθε περίπτωση τον αριθμό των κελιών που δεν συμφωνούν με την τελική κατάσταση.

β) Ποιός αλγόριθμος πρέπει να χρησιμοποιηθεί ώστε να βρούμε β1) μια οποιαδήποτε λύση, β2) όλες τις λύσεις, β3) τη συντομότερη, β4) τη βέλτιστη λύση. Δικαιολογείστε τις απαντήσεις σας.

(β1) μία λύση:

Κατά βάθος (depth-first) (Είναι συνήθως υπολογιστικά πιο «φθηνός» και πιο γρήγορος στο να βρίσκει μια λύση)

(β2) όλες τις λύσεις:

Κατά πλάτος (breadth-first) (Βρίσκει όλες τις λύσεις με ασφάλεια και χωρίς χρήση ευρετικών.)

Επαναληπτική εκβάθυνση (iterative-deepening) (Βρίσκει όλες τις λύσεις με ασφάλεια, σε περισσότερο χρόνο από τον κατά πλάτος, αλλά με λιγότερη χρήση μνήμης.)

(β3) συντομότερη λύση = λιγότερα βήματα:

Κατά πλάτος (Βρίσκει πάντα τη συντομότερη λύση, δηλ. αυτή με τα λιγότερα βήματα, που δε σημαίνει όμως ότι είναι και η βέλτιστη. Στην περίπτωσή μας όμως, λόγω σταθερού κόστους βρίσκει και την βέλτιστη.)

(β4) βέλτιστη λύση = μικρότερο κόστος :

A* (Εγγυάται εύρεση της βέλτιστης λύσης. Δεδομένου όμως ότι το κόστος είναι σταθερό, ουσιαστικά μετατρέπεται σε αλγόριθμο βέλτιστου κόμβου (best-first).

γ) Εφαρμόστε τα τρία πρώτα βήματα του αλγορίθμου που προτείνατε για το β4, σχεδιάζοντας τα αντίστοιχα επίπεδα στο δέντρο καταστάσεων.

Εφαρμογή των τριών πρώτων βήματων του αλγορίθμου Α*

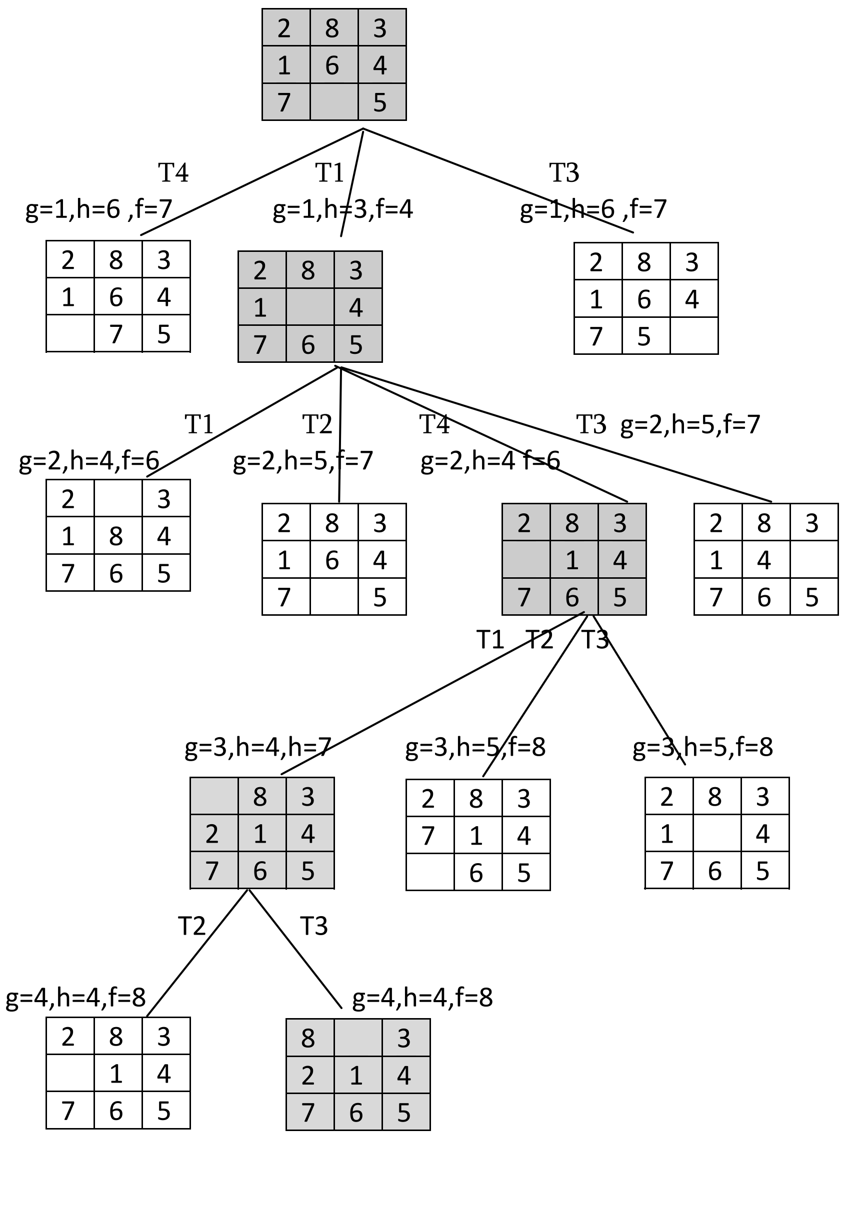
Ο αλγόριθμος Α* είναι συνδυασμός των αλγορίθμων διακλάδωση και δέσμευση (ανάπτυξη του κόμβου της συντομότερης τρέχουσας διαδρομής f(n)=g(n)) και βέλτιστου κόμβου (ανάπτυξη του καλύτερου κόμβου f(n)=h(n)).

Η συνάρτηση f(n) στον αλγόριθμο Α* είναι f(n) = g(n) + h(n). Για εφαρμογή του αλγορίθμου έχουμε ότι

α) Η κατάσταση η οποία είναι ίδια με κάποια προηγούμενη διαγράφετε.

β)Στην περίπτωση που δύο ή περισσότερες καταστάσεις έχουν την ίδια εκτίμηση f τότε επιλέγετε το δεξιότερο.

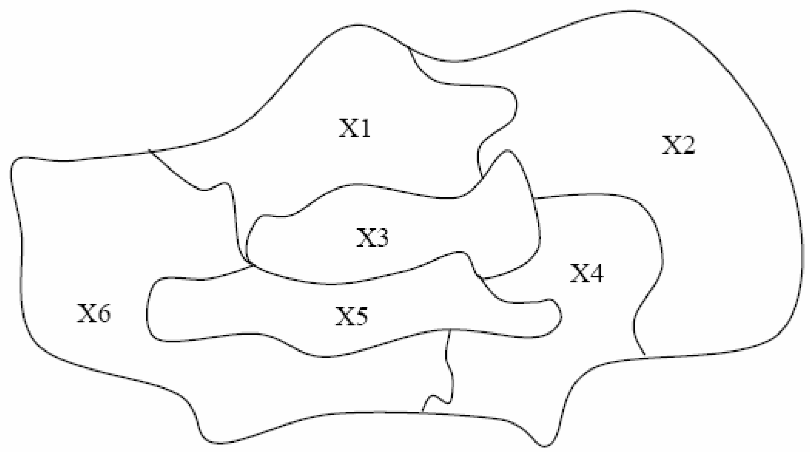
γ) Σε περίπτωση που υπάρχουν δύο ή περισσότερες ίδιες καταστάσεων σε διαφορετικά επίπεδα διαγράφουμε αυτήν που βρίσκεται ποιο μακριά από την αρχική.



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ & ΠΛΗΡΟΦΟΡΙΚΗΣ
ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ
2008-2009
4η Σειρά Ασκήσεων

ΑΠΑΝΤΗΣΕΙΣ

Δίνεται το εξής πρόβλημα: Έχετε ένα επίπεδο χάρτη (όπως αυτόν του σχήματος) που περιέχει έξη (6) διαφορετικές περιοχές (π.χ. χώρες) και θέλετε να τον χρωματίσετε χρησιμοποιώντας μόνο τέσσερα διαφορετικά (4) χρώματα (κόκκινο, κίτρινο, πράσινο, μπλε), έτσι ώστε να μην υπάρχουν δύο γειτονικές περιοχές με το ίδιο χρώμα.



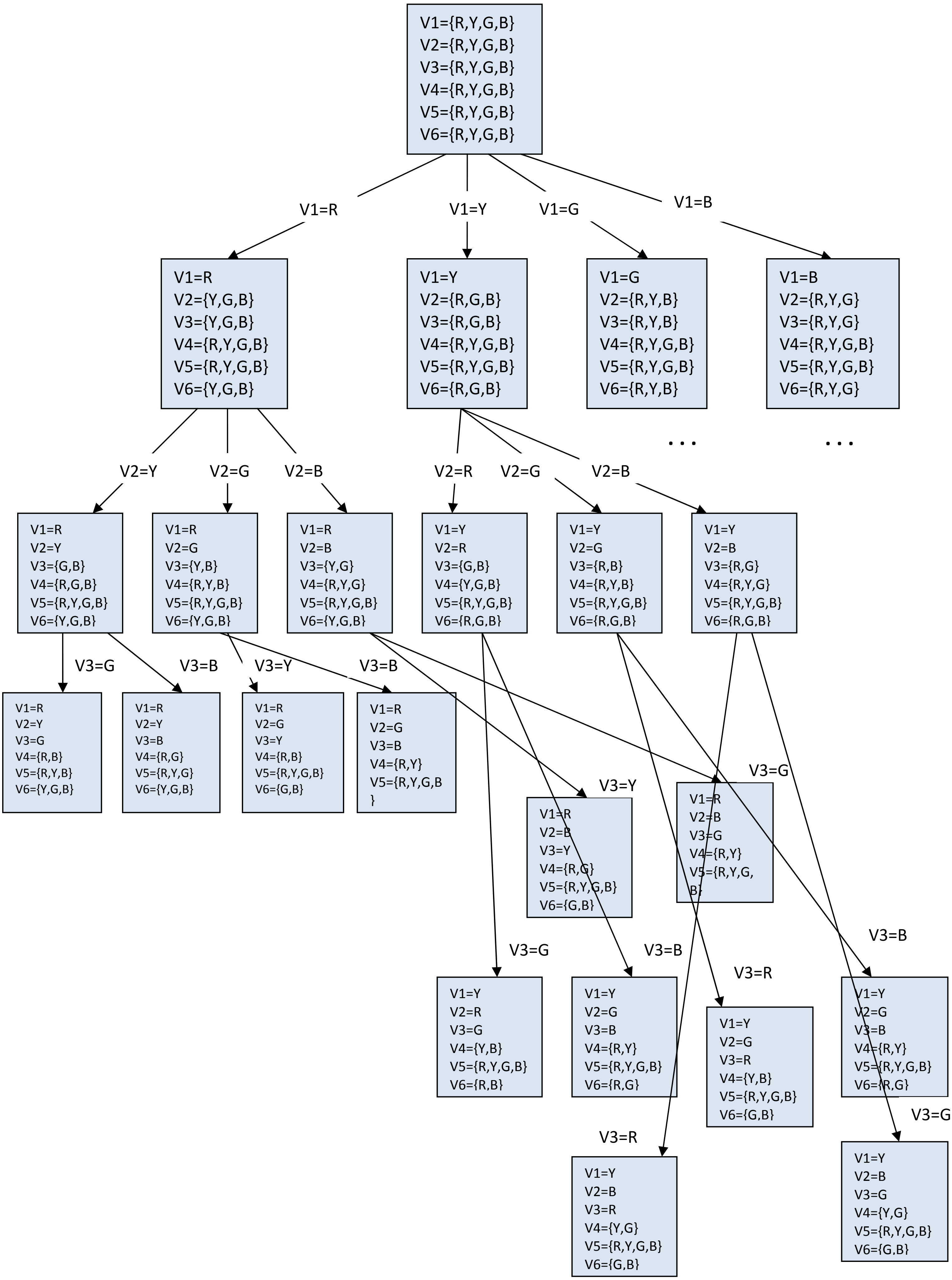
Ζητείται να περιγραφεί σαν πρόβλημα ικανοποίησης περιορισμών, δηλ. να οριστούν οι μεταβλητές, τα πεδία τιμών τους και οι περιορισμοί μεταξύ των μεταβλητών.

<u>Αναπαράσταση Προβλήματος:</u>
Μεταβλητές: V1, V2, V3, V4, V5 ,V6
Πεδία Τιμών : D1≡D2≡D3≡D4≡D5≡D6≡={R,Y,G,B}

Ο μόνος περιορισμός που μας δίνεται απ' το πρόβλημα είναι ότι δύο γειτονικές μεταβλητές (περιοχές στον χάρτη) δεν μπορούν να έχουν ταυτόχρονα την ίδια τιμή (χρώμα).
Επομένως θα πρέπει να ικανοποιούνται οι ακόλουθοι περιορισμοί:

V1 ≠ V2 (C1)	V1 ≠ V6 (C3)	V3 ≠ V5 (C7)	V5 ≠ V4 (C10)
V1 ≠ V3 (C2)	V6 ≠ V3 (C4)	V3 ≠ V2 (C8)	V2 ≠ V4 (C11)
V1 ≠ V6 (C3)	V6 ≠ V5 (C5)	V3 ≠ V4 (C9)	

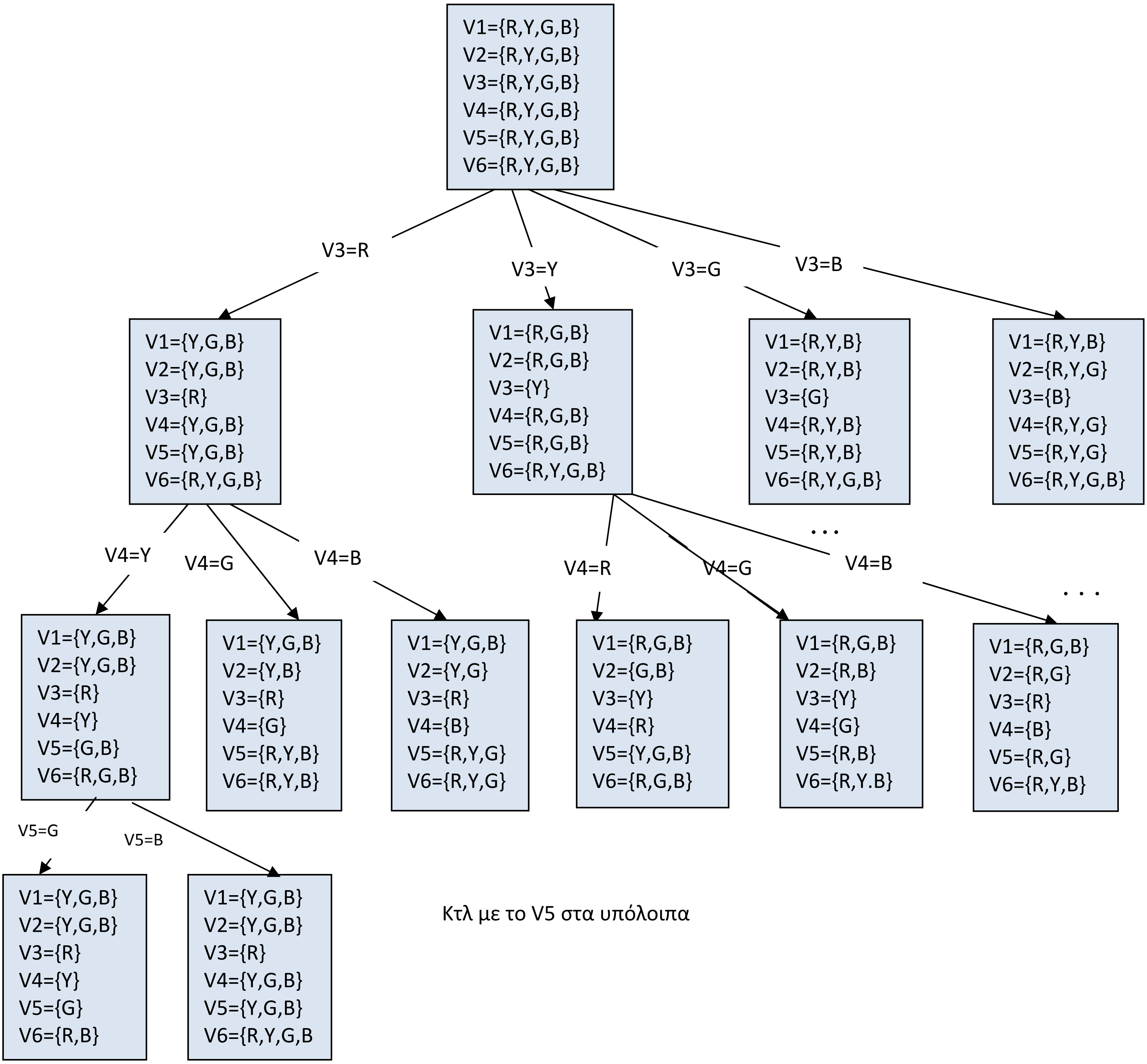
(1) Εφαρμόστε ένα αλγόριθμο διατήρησης συνέπειας τόξου, συνδυάζοντας τον κλασσικό αλγόριθμο αναζήτησης 'πρώτα σε βάθος' με τον αλγόριθμο ελέγχου συνέπειας AC-3. Σχεδιάστε τα τρία πρώτα επίπεδα του δένδρου αναζήτησης.



V1={R,Y,G,B}
V2={R,Y,G,B}
V3={R,Y,G,B}
V4={R,Y,G,B}
V5={R,Y,G,B}
V6={R,Y,G,B}
V1=R
V1=Y
V1=G
V1=B
V1=R
V2={Y,G,B}
V3={Y,G,B}
V4={R,Y,G,B}
V5={R,Y,G,B}
V6={Y,G,B}
V1=Y
V2={R,G,B}
V3={R,G,B}
V4={R,Y,G,B}
V5={R,Y,G,B}
V6={R,G,B}
V1=G
V2={R,Y,B}
V3={R,Y,B}
V4={R,Y,G,B}
V5={R,Y,G,B}
V6={R,Y,B}
V1=B
V2={R,Y,G}
V3={R,Y,G}
V4={R,Y,G,B}
V5={R,Y,G,B}
V6={R,Y,G}
. . .
. . .
V2=Y
V2=G
V2=B
V2=R
V2=G
V2=B
V1=R
V2=Y
V3={G,B}
V4={R,G,B}
V5={R,Y,G,B}
V6={Y,G,B}
V1=R
V2=G
V3={Y,B}
V4={R,Y,B}
V5={R,Y,G,B}
V6={Y,G,B}
V1=R
V2=B
V3={Y,G}
V4={R,Y,G}
V5={R,Y,G,B}
V6={Y,G,B}
V1=Y
V2=R
V3={G,B}
V4={Y,G,B}
V5={R,Y,G,B}
V6={R,G,B}
V1=Y
V2=G
V3={R,B}
V4={R,Y,B}
V5={R,Y,G,B}
V6={R,G,B}
V1=Y
V2=B
V3={R,G}
V4={R,Y,G}
V5={R,Y,G,B}
V6={R,G,B}
V3=G
V3=B
V3=Y
V3=B
V1=R
V2=Y
V3=G
V4={R,B}
V5={R,Y,B}
V6={Y,G,B}
V1=R
V2=Y
V3=B
V4={R,G}
V5={R,Y,G}
V6={Y,G,B}
V1=R
V2=G
V3=Y
V4={R,B}
V5={R,Y,G,B}
V6={G,B}
V1=R
V2=G
V3=B
V4={R,Y}
V5={R,Y,G,B
}
V3=Y
V1=R
V2=B
V3=Y
V4={R,G}
V5={R,Y,G,B}
V6={G,B}
V3=G
V1=R
V2=B
V3=G
V4={R,Y}
V5={R,Y,G,
B}
V3=G
V1=Y
V2=R
V3=G
V4={Y,B}
V5={R,Y,G,B}
V6={R,B}
V3=B
V1=Y
V2=G
V3=B
V4={R,Y}
V5={R,Y,G,B}
V6={R,G}
V3=R
V1=Y
V2=G
V3=R
V4={Y,B}
V5={R,Y,G,B}
V6={G,B}
V3=B
V1=Y
V2=G
V3=B
V4={R,Y}
V5={R,Y,G,B}
V6={R,G}
V3=R
V1=Y
V2=B
V3=R
V4={Y,G}
V5={R,Y,G,B}
V6={G,B}
V3=G
V1=Y
V2=B
V3=G
V4={Y,G}
V5={R,Y,G,B}
V6={G,B}

(2) Το ίδιο με το (1), μόνο που θα πρέπει να χρησιμοποιήσετε ως κλασσικό αλγόριθμο τον Best-First Search, αφού ορίσετε κατάλληλο ευρετικό.

Θα χρησιμοποιήσουμε τον κλασσικό αλγόριθμο, Best-First Search, ο οποίος έχει καλύτερα αποτελέσματα από τον DFS λόγω του ότι είναι ευριστικός αλγόριθμος αναζήτησης.
Η ευριστική συνάρτηση αφορά την επιλογή της μεταβλητής για ανάθεση τιμής στο επόμενο βήμα. Στηρίζεται:
1. στην αρχή της συντομότερης αποτυχίας(επιλογή μεταβλητής με το μικρότερο πεδίο τιμών) και
2. στην αρχή της πιο περιορισμένης μεταβλητής(επιλογή της μεταβλητής που συμμετέχει στους περισσότερους περιορισμούς σε περίπτωση ισοδύναμων πεδίων τιμών).



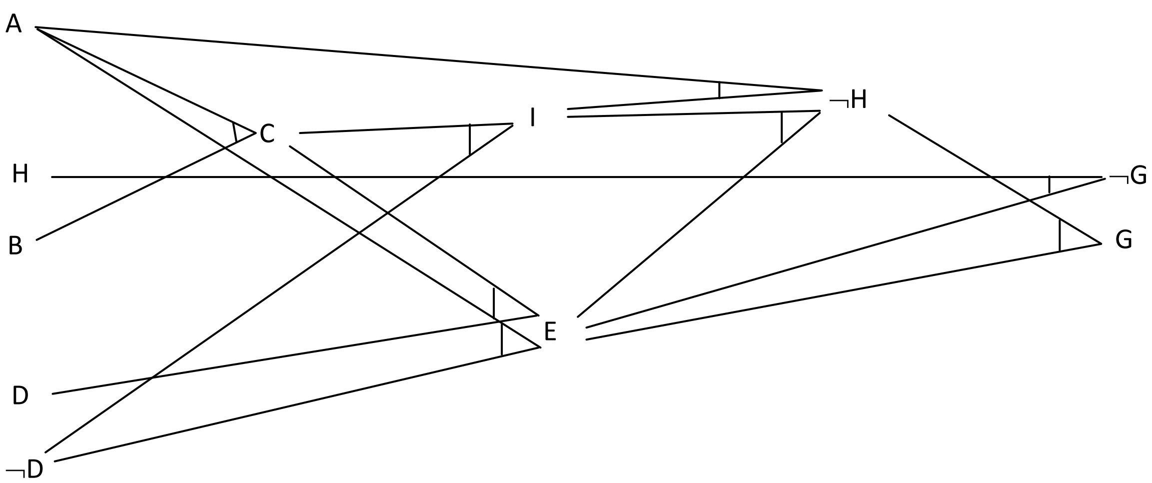
ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ & ΠΛΗΡΟΦΟΡΙΚΗΣ
ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ
2008-2009
5η Σειρά Ασκήσεων

ΑΠΑΝΤΗΣΕΙΣ

1. Δίνεται η παρακάτω βάση κανόνων:

R1: **if** A **and** B **then** C
R2: **if** C **and** D **then** E
R3: **if** A **and** I **then** ¬H
R4: **if** A **and** ¬D **then** E
R5: **if** C **and** ¬D **then** I
R6: **if** E **and** I **then** ¬H
R7: **if** E **and** H **then** ¬G
R8: **if** E **and** ¬H **then** G

(α) Σχεδιάστε το δίκτυο της βάσης κανόνων.

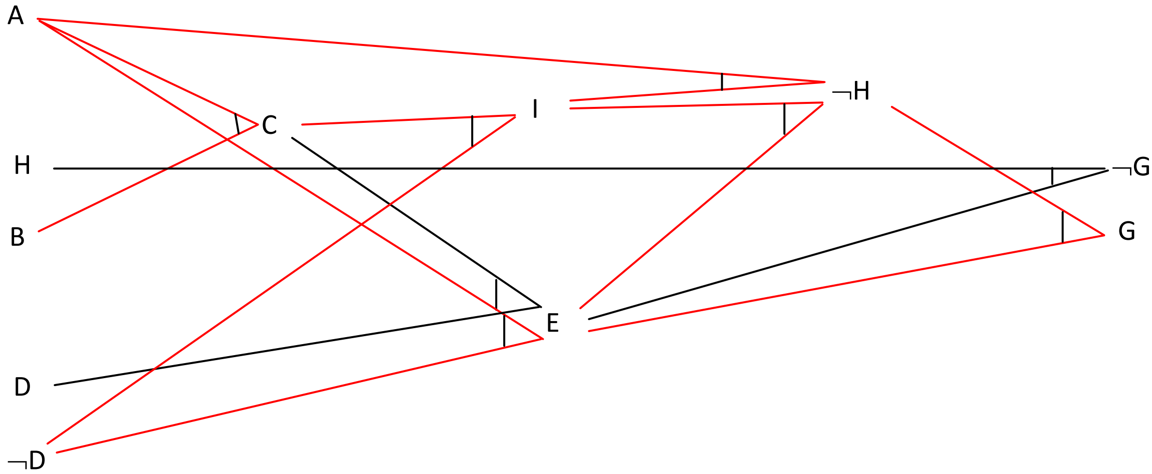


(β) Ζητείται να εξαχθεί το G, αν

- το αρχικό περιεχόμενο της μνήμης εργασίας είναι WM = {A, B, ¬D, ¬H}
- χρησιμοποιηθεί αλυσίδωση προς τα εμπρός (forward chaining)
- ο πρώτος υποψήφιος κανόνας που συναντάται πυροδοτείται (σειρά αναγραφής)
- ο ίδιος κανόνας μόνο μια φορά πυροδοτείται

Περιγράψτε τα βήματα της εξαγωγής και αποτυπώστε τις εξαγωγές και στο δίκτυο που σχεδιάσατε στο (α).

	WM = { A, B, ¬D, ¬H }
R1->C	WM = { A, B, ¬D, ¬H, C }
R4->E	WM = { A, B, ¬D, ¬H, C, E }
R5->I	WM = { A, B, ¬D, ¬H, C, E, I }
R3-> ¬H	WM = { A, B, ¬D, ¬H, C, E, I}
R6-> ¬H	WM = { A, B, ¬D, ¬H, C, E, I}
R8->G	WM = { A, B, ¬D, ¬H, C, E, I, G }



(γ) Το ίδιο με το (β) όπου όμως θα χρησιμοποιήσετε ως στρατηγική ελέγχου την προσφατότητα και δευτερευόντως τη σειρά αναγραφής (η μη πυροδότηση του ίδιου κανόνα εξακολουθεί να ισχύει). Υπάρχει διαφορά; Εξηγείστε.

WM = { A, B, ¬D, ¬H } (ΣΣ: Σύνολο Σύγκρουσης)

R1->C WM = { A, B, ¬D, ¬H, C } (την πρώτη φορά επιλέγεται ο πρώτος υποψήφιος κανόνας που συναντάμε)

ΣΣ = {R4, R5} και επιλέγεται ο R5 διότι οι συνθήκες του ταιριάζουν με πιο πρόσφατα γεγονότα στη WM

R5->I WM = { A, B, ¬D, ¬H, C, I }

ΣΣ = {R3, R4} και επιλέγεται ο R3 διότι οι συνθήκες του ταιριάζουν με πιο πρόσφατα γεγονότα στη WM

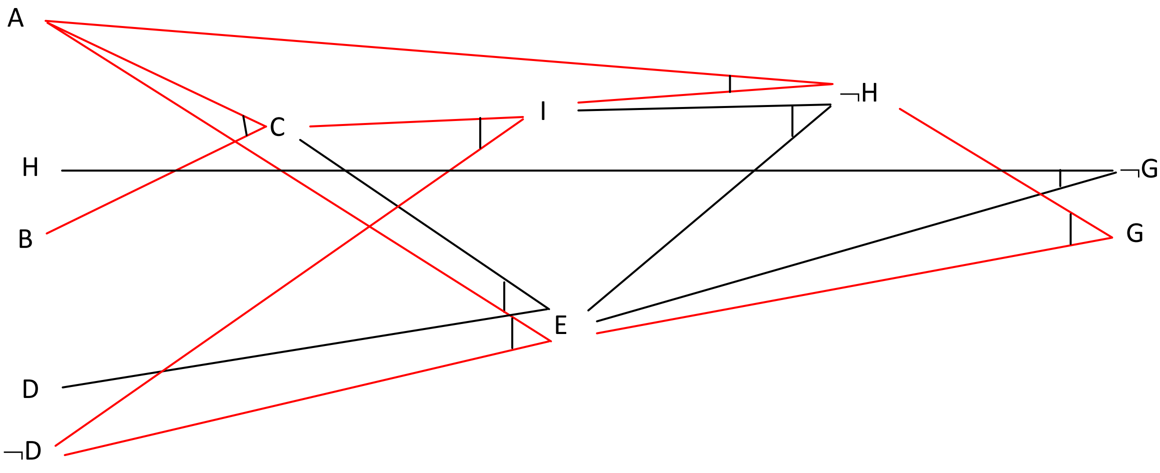
R3-> ¬H WM = { A, B, ¬D, ¬H, C, I, ¬H}

ΣΣ = {R4} και επιλέγεται ο R4 διότι είναι ο μοναδικός υποψήφιος κανόνας

R4->E WM = { A, B, ¬D, ¬H, C, I, ¬H, E }

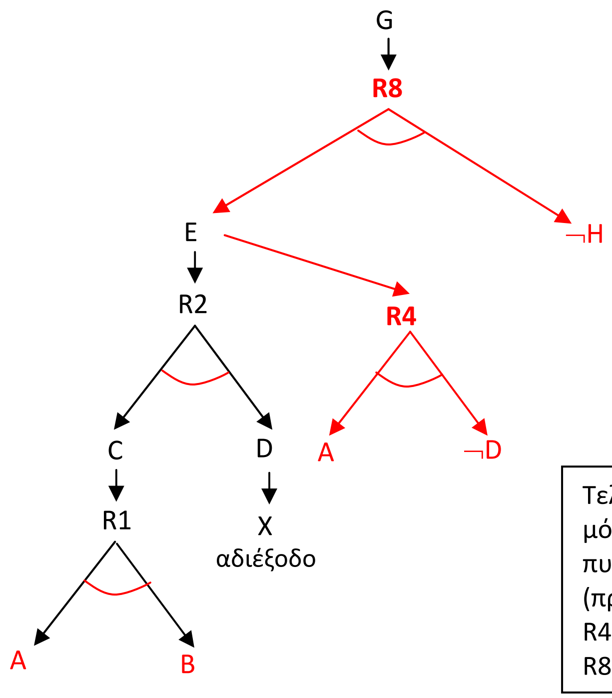
ΣΣ = {R6, R8} και επιλέγεται ο R8 διότι οι συνθήκες του ταιριάζουν με πιο πρόσφατα γεγονότα στη WM

R8->G WM = { A, B, ¬D, ¬H, C, I, ¬H, E, G }



Παρατηρούμε ότι τώρα πυροδοτήθηκε ένας κανόνας λιγότερο, δηλ. φτάσαμε στο στόχο γρηγορότερα (σε λιγότερα βήματα).

(δ) Το ίδιο με το (β), μόνο που θα χρησιμοποιήσετε ανάστροφη αλυσίδωση αντί ορθής. Επισημάνετε τις διαφορές με το (β).



- Το διπλανό δέντρο προχωρά από αριστερά προς τα δεξιά μέχρι να ικανοποιηθούν οι συνθήκες των αντίστοιχων κανόνων.
- Στο αδιέξοδο επιστρέφει και επιλέγει τον επόμενο κανόνα (R4)
- Τα κόκκινα γράμματα υποδηλώνουν γεγονότα που υπάρχουν στη μνήμη εργασίας και κανόνες που πυροδοτούνται

Τελικά τρεις μόνο κανόνες εξετάζονται και δύο μόνο από αυτούς (R4, R8) χρειάζεται να πυροδοτηθούν για να βρεθεί η λύση (προχωρώντας αντίστροφα):
R4 -> E
R8 -> G

2. Δίνονται οι παρακάτω πέντε κανόνες που χρησιμοποιούν συντελεστές βεβαιότητας και αποτελούν τη βάση κανόνων ενός συστήματος.

R1 **if shape is round** **then fruit is orange (0.3)**	**R4** **if shape is round** **and color is yellow** **then fruit is apricot (0.6)**
R2 **If shape is round** **then fruit is apricot (0.2)**	**R5** **if shape is round** **and color is yellow** **and size is small** **then fruit is apricot (0.8)**
R3 **if shape is round** **and surface is weasand** **then fruit is orange (0.8)**	

Αν δοθούν τα εξής δεδομένα διαδοχικά: “shape is round”, “color is yellow (0.6)”, “size is small (0.7)” και “surface is weasand (0.9)” περιγράψτε τα βήματα εξαγωγής συμπερασμάτων. Ποιο είναι το τελικό συμπέρασμα;

“shape is round”
“color is yellow (0.6)”
“size is small (0.7)”
“surface is weasand (0.9)”

1. Λόγω του “shape is round” ενεργοποιούνται οι R1 και R2 οπότε
   ME = { fruit is orange (0.3)
   fruit is apricot (0.2) }

2. Λόγω του “color is yellow (0.6)” ενεργοποιείται ο R4: cfR4 = min(cfe1, cfe2) * cfh = min(1.0, 0.6) * 0.6 = 0.36, οπότε
   ME = { fruit is orange (0.3)
   fruit is apricot (0.2) }
   fruit is apricot (0.36) }
   Επειδή έχουμε δύο ίδια γεγονότα-συμπεράσματα από δύο διαφορετικούς κανόνες πρέπει να συνδυάσουμε τους συντελεστές βεβαιότητας. Επειδή είναι και οι δύο θετικοί, εφαρμόζουμε την σχέση:
   cfR2R4 = cfR2+cfR4-cfR2*cfR4 = 0.2+0.36-0.2*0.36 = 0.488
   Επομένως η ME γίνεται:
   ME = { fruit is orange (0.3)
   fruit is apricot (0.488) }

3. Λόγω του “size is small (0.7)” ενεργοποιείται ο R5: cfR5 = min(cfe1, cfe2, cfe3) * cfh = min(1.0, 0.6, 0.7) * 0.8 = 0.48, οπότε
   ME = { fruit is orange (0.3)
   fruit is apricot (0.488) }
   fruit is apricot (0.48) }
   Πάλι έχουμε ίδια συμπεράσματα από διαφορετικούς κανόνες, οπότε
   cfR2R4R5 = cfR2R4+cfR5-cfR2R4*cfR5 = 0.488+0.48-0.488*0.48 = 0.734
   Επομένως η ME γίνεται:
   ME = { fruit is orange (0.3)
   fruit is apricot (0.734) }

4. Λόγω του “surface is weasand (0.9)” ενεργοποιείται ο R3: cfR3 = min(cfe1, cfe2) * cfh = min(1.0, 0.9) * 0.8 = 0.72, οπότε
   ME = { fruit is orange (0.3)
   fruit is apricot (0.734) }
   fruit is orange (0.72) }
   Πάλι έχουμε ίδια συμπεράσματα από διαφορετικούς κανόνες, οπότε
   cfR1R3 = cfR1+cfR3-cfR1*cfR3 = 0.3+0.72-0.3*0.72 = 0.804
   Επομένως η ME γίνεται:
   ME = { fruit is orange (0.804)
   fruit is apricot (0.734) }

Επομένως, το υπ’ όψιν φρούτο είναι με μεγαλύτερη βεβαιότητα (0.804) πορτοκάλι (orange) παρά (0.734) βερίκοκο (apricot).

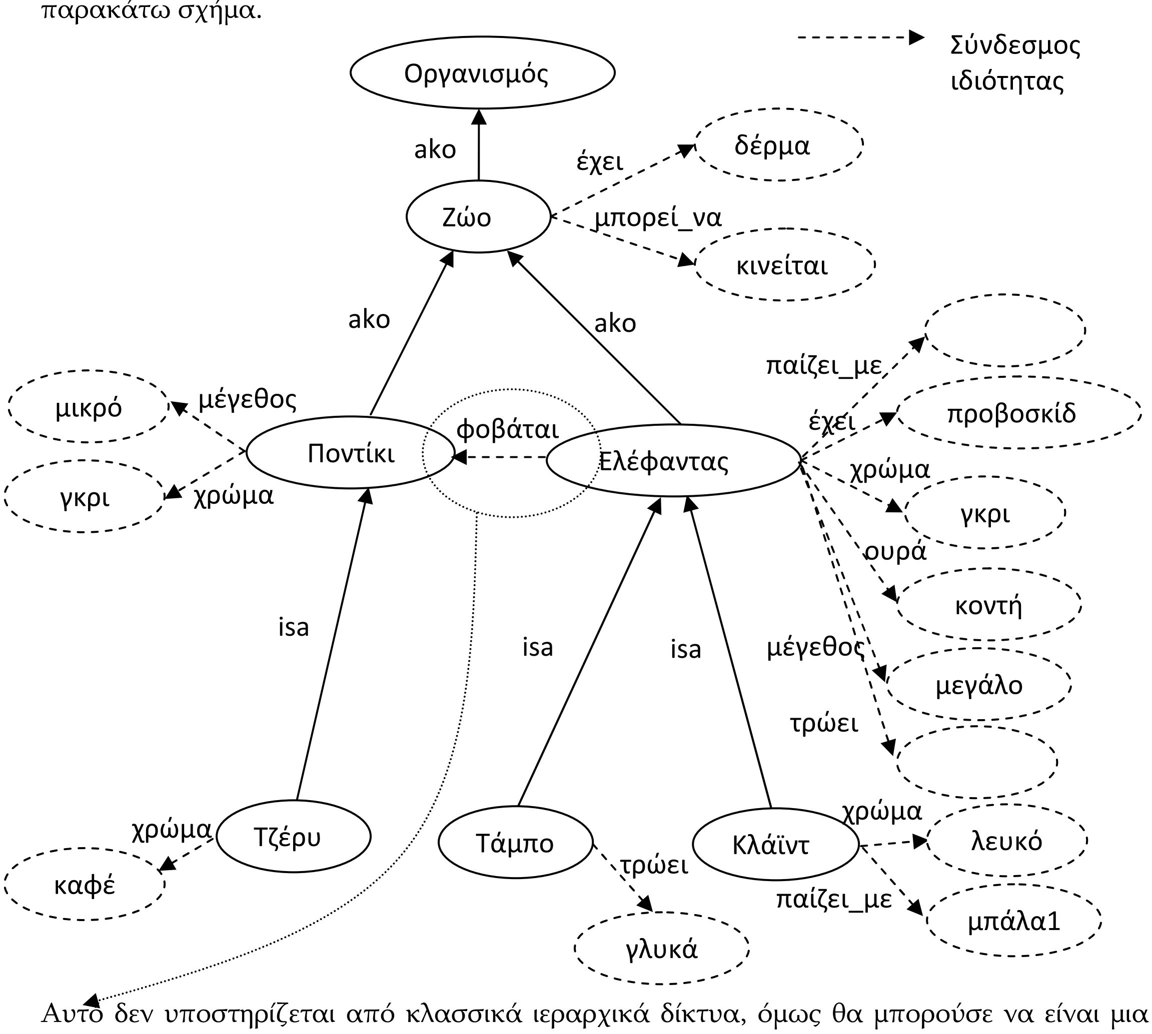
ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ & ΠΛΗΡΟΦΟΡΙΚΗΣ
ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ
2008-2009
6η Σειρά Ασκήσεων

ΑΠΑΝΤΗΣΕΙΣ

1. Δίνεται η εξής περιγραφή: «Τα ζώα είναι οργανισμοί που έχουν δέρμα και κινούνται. Οι ελέφαντες είναι ζώα μεγάλου μεγέθους που διαθέτουν προβοσκίδα, κοντή ουρά και είναι συνήθως χρώματος γκρι. Τα ποντίκια είναι ζώα μικρού μεγέθους που διαθέτουν μακριά ουρά και είναι κι' αυτά συνήθως χρώματος γκρι. Οι ελέφαντες φοβούνται τα ποντίκια. Ο Τάμπο και ο Κλάιντ είναι ελέφαντες. Όμως, ο Κλάιντ είναι λευκός ελέφαντας. Ο Τάμπο συνηθίζει να τρώει γλυκά, ενώ ο Κλάιντ να παίζει με μια (συγκεκριμένη) μπάλα. Ο Τζέρυ είναι ένα ποντίκι χρώματος καφέ. »

α. Να αναπαρασταθεί η παραπάνω γνώση με ένα σημαντικό δίκτυο.

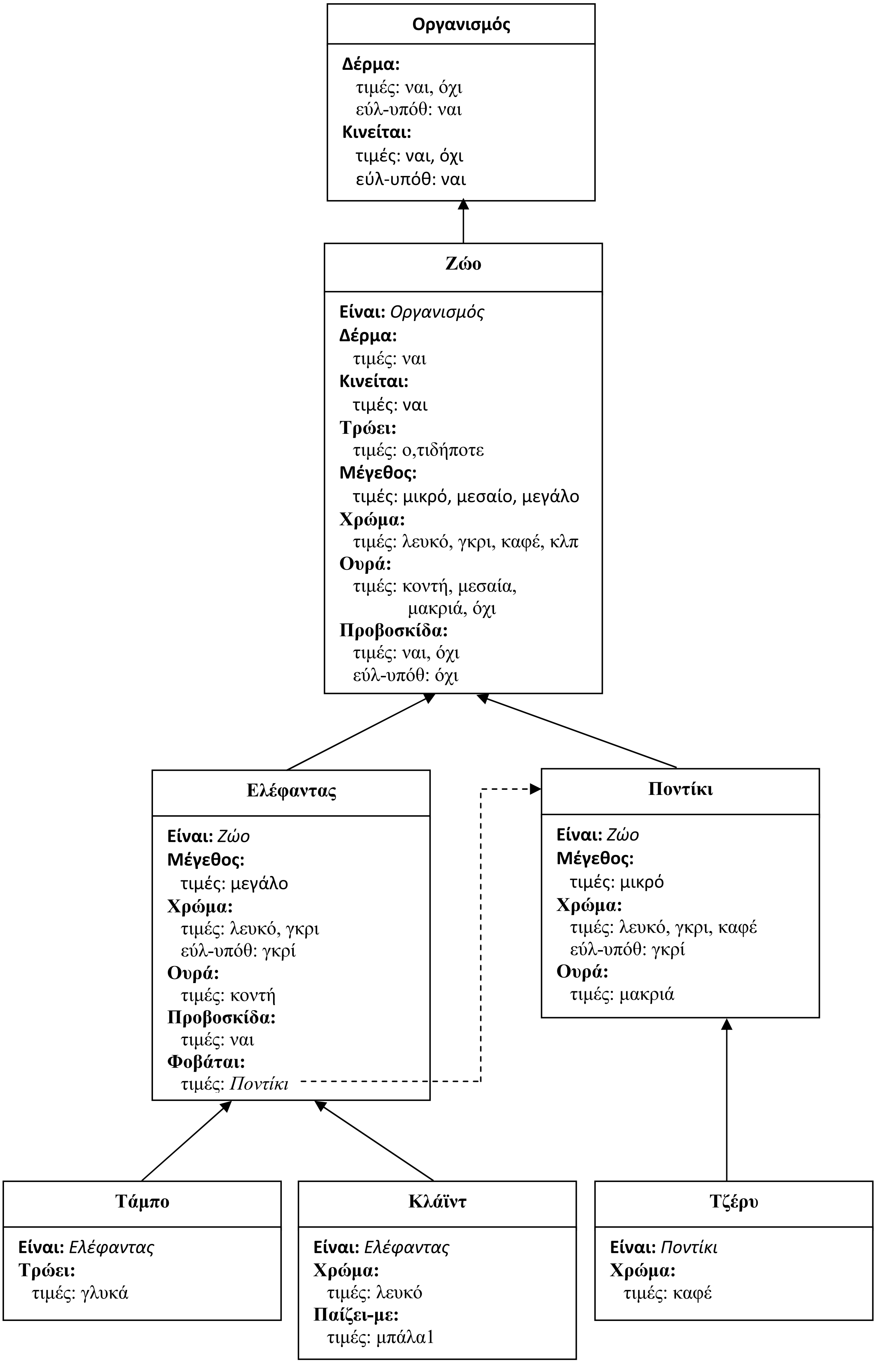
Επιλέγουμε ένα ιεραρχικό σημαντικό δίκτυο για την αναπαράσταση, η οποία φαίνεται στο παρακάτω σχήμα.



Αυτό δεν υποστηρίζεται από κλασσικά ιεραρχικά δίκτυα, όμως θα μπορούσε να είναι μια επέκτασή τους (δηλ. σύνδεσμος ιδιότητας να δείχνει σε κόμβο της ιεραρχίας).

β. Να αναπαρασταθεί η παραπάνω γνώση με πλαίσια.

Η αναπαράσταση με πλαίσια φαίνεται στο επόμενο σχήμα.

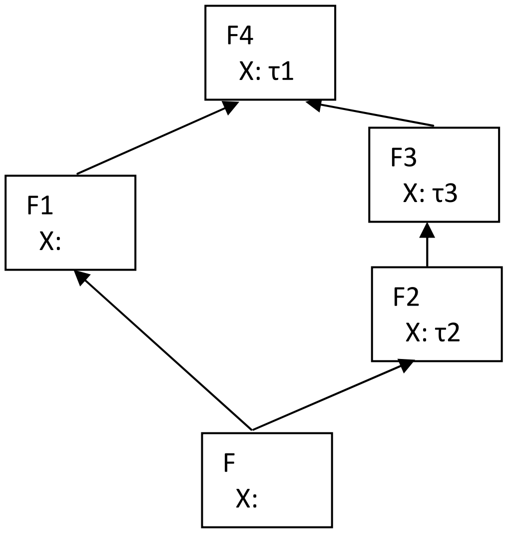


Και δω η σύνδεση μια τιμής με ένα πλαίσιο δεν είναι κάτι συνηθισμένο, αλλά συναντάται πιο συχνά σε γλώσσες πλαισίων απ' ότι σε σημαντικά δίκτυα.

γ. Εξηγείστε πώς βρίσκεται η απάντηση στην ερώτηση «Ποιόν φοβάται ο Τάμπο;» και στις δύο παραπάνω αναπαραστάσεις.

Το ερώτημα «Ποιόν φοβάται ο Τάμπο;» μεταφέρεται ως ερώτημα για την τιμή της ιδιότητας «φοβάται» στον κόμβο/πλαίσιο 'Τάμπο'. Επειδή δεν υπάρχει η απάντηση τοπικά στο πλαίσιο αυτό, χρησιμοποιείται η διαδικασία της κληρονόμησης ιδοτήτων-τιμών από τους κόμβους/πλαίσια που βρίσκονται υψηλότερα στην ιεραρχία. Οπότε αναζητείται η τιμή της ιδιότητας 'φοβάται' στον κόμβο/πλαίσιο 'Ελέφαντας'. Η τιμή της ιδιότητας αυτής οδηγεί σε ένα άλλο κόμβο/πλαίσιο με όνομα 'Ποντίκι'. Αυτός ο σύνδεσμος επιστρέφει τα ονόματα των στιγμιοτύπων της κλάσης 'Ποντίκι', που εδώ είναι μόνο το 'Τζέρυ'. Αυτό επιστρέφεται και ως απάντηση στο ερώτημα.

2. Στην παρακάτω ιεραρχία πλαισίων να εξηγήσετε τι απάντηση θα δώσει ο Βασισμένος στην Απόσταση Συλλογισμού αλγόριθμος στην ερώτηση για την τιμή του χαρακτηριστικού X του πλαισίου F.



1. Βρίσκουμε κατ' αρχήν όλες τις τιμές του 'X' προχωρώντας π.χ. κατά πλάτος και τις αποθηκεύουμε στη λίστα VALUES:

   VALUES = [τ2, τ1, τ3]
2. Για κάθε τιμή στην VALUES εξετάζουμε αν κάποια από τις υπόλοιπες προήλθε από πλαίσιο που βρίσκεται σε μικρότερη απόσταση συλλογισμού από το πλαίσιο που προήλθε η τρέχουσα εξεταζόμενη τιμή, ως προς υπό ερώτηση πλαίσιο F. Αν υπάρχει τέτοια, τότε διαγράφουμε την τρέχουσα εξεταζόμενη τιμή. Τώρα, ξεκινώντας τη διαδικασία ελέγχου οι τ1 και τ3 διαγράφονται διότι βρίσκονται σε πλαίσια (F3 και F4 αντίστοιχα) που έχουν μεγαλύτερη συλλογιστική απόσταση ως προς το F από το F2 (που βρίσκεται η τ2), οπότε:

   VALUES = [τ2]

   και η απάντηση είναι τ2.

Έγινε μια αλλαγή εδώ για καθαρότερα αποτελέσματα

3. Δίνονται οι παρακάτω ασαφείς κανόνες:

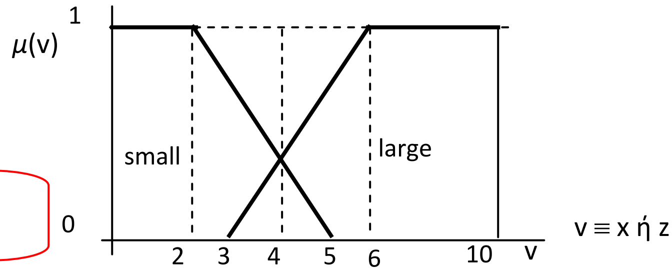
R1	R2	R3
if x is large and y is high then z is small	if x is small and y is low then z is small	if x is small and y is high then z is large

όπου x, y, z λεκτικές μεταβλητές, με y $\in$[0, 5] και x, z $\in$[0, 10]. Οι λεκτικές τιμές (ασαφή σύνολα) και για τις τρεις μεταβλητές καθώς και οι συναρτήσεις συμμετοχής δίνονται παρακάτω:

$\mu_{low}(y)=1-(y/3)$ $(0\le y\le 3)$

$\mu_{high}(y)=(y-1)/4$ $(1\le y\le 5)$

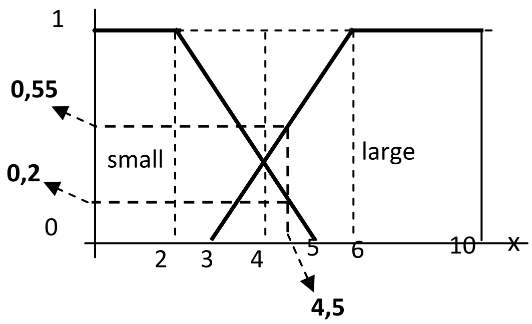
Έγινε μια αλλαγή και δώ για τον ίδιο λόγο.



Αν οι τιμές εισόδου είναι x1= 4,5 και y1=2,7 να βρεθεί η έξοδος z1. Για τη σύνθεση βαθμών συμμετοχής μέσω and να χρησιμοποιηθεί η μέθοδος prod.

1. Ασαφοποίηση

Για το x1 μέσω των γραφικών παραστάσεων.



$\mu_{small}(x1) = 0{,}2$

$\mu_{large}(x1) = 0{,}55$

Για το y1 μέσω των αναλυτικών τύπων που δίνονται.

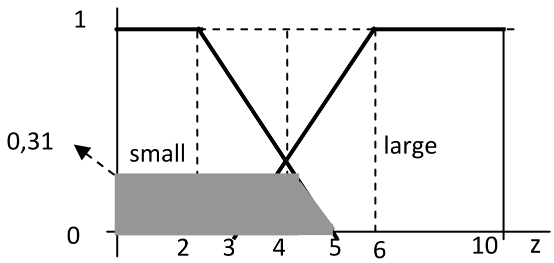
$\mu_{low}(y1) = 1\text{-}(y1/3) = 1\text{-}(2{,}7/3) = 1\text{-}0{,}9 = 0{,}1$

$\mu_{high}(y1) = (y1\text{-}1)/3 = (2{,}7\text{-}1)/3 = 1{,}7/3 = 0{,}57$

2. Εκτίμηση κανόνων

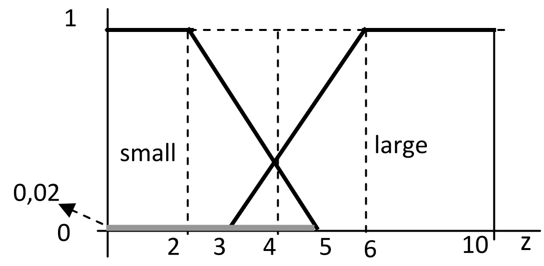
R1
if x is large (0,55)
and y is high (0,57)
then z is small (0,31)

0,55 * 0,57 = 0,31



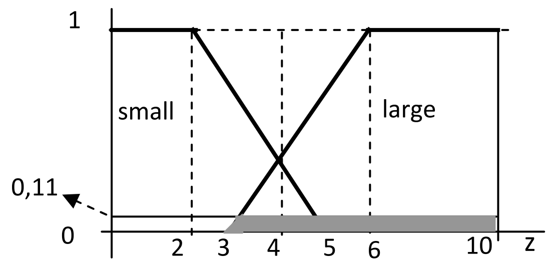
R2
if x is small (0,2)
and y is low (0,1)
then z is small (**0,02**)

0,2 * 0,1 = 0,02



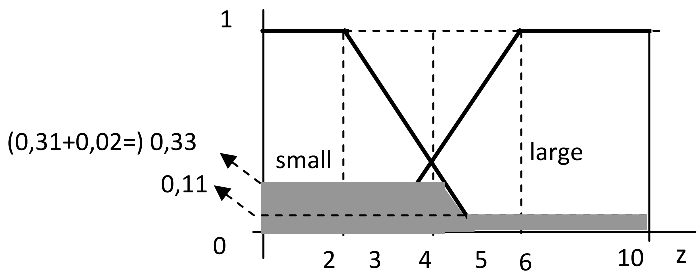
R3
if x is small (0,2)
and y is high (0,57)
then z is large (**0,11**)

0,2 * 0,57 = 0,11



3. Συνάθροιση

Για τις εξόδους των R1 και R2 δεν προσθέτουμε τις εξόδους ακριβώς (κι' αυτό θα μπορούσε να γίνει), αλλά προσθέτουμε τους βαθμούς συμμετοχής και «κόβουμε» την καμπύλη large σε κείνο το ύψος.



4. Αποασαφοποίηση

z = [(0+1+2+3+4) * 0,33 + (5+6+7+8+9+10) * 0,11]/
(4*0,33+ 6 * 0,11)
= 8,25/1,98 = 4,17

z1 = 4,17