ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΘΕΜΕΛΙΩΔΗ ΘΕΜΑΤΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

(FOCS)

(2020-2021)

3η Σειρά Ασκήσεων

Ονοματεπώνυμο:

- Χρήστος Τσούφης

Αριθμός Μητρώου:

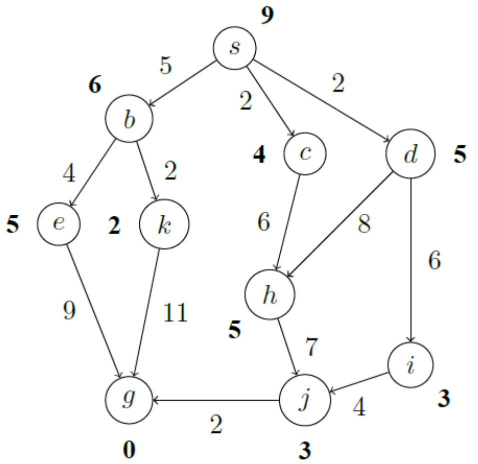
- 03117176

Στοιχεία Επικοινωνίας:

- el17176@mail.ntua.gr

1η Άσκηση (Αλγόριθμοι αναζήτησης λύσης - Άσκηση)

Δίνεται ο παρακάτω χώρος αναζήτησης, όπου s είναι η αρχική και g η τελική κατάσταση. Οι αριθμοί δίπλα σε κάθε ακμή αντιπροσωπεύουν την πραγματική απόσταση των κόμβων που συνδέει η ακμή, και οι αριθμοί δίπλα σε κάθε κατάσταση (με έντονα γράμματα) αντιπροσωπεύουν την τιμή της ευριστικής εκτίμησης της απόστασης μέχρι την τελική κατάσταση.



1. *Εκτελέστε τον αλγόριθμο αναρρίχησης λόφων και τον αλγόριθμο Α∗ για το παραπάνω πρόβλημα.*

Ο αλγόριθμος για τον **Hill Climbing**:

Βήμα 1: Όρισε τον τρέχοντα κόμβο ως τη ρίζα του δένδρου.

Βήμα 2: Μέχρι που ο τρέχων κόμβος δεν είναι κόμβος στόχος, εκτέλεσε:

Βήμα 2.α: Βρες τα παιδιά του τρέχοντος κόμβου, και στη συνέχεια βρες αυτό με την ελάχιστη υπολογιζόμενη υπόλοιπη απόσταση από το στόχο.

Βήμα 2.β: Εάν ο τρέχων κόμβος δεν έχει παιδιά ή το παιδί που βρέθηκε στο βήμα 2.α έχει μεγαλύτερη τιμή ευριστικής πήγαινε στο Βήμα 3.

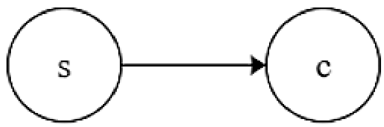
Βήμα 2.γ: Όρισε τον κόμβο που βρέθηκε στο Βήμα 2.α ως τρέχων κόμβο.

Βήμα 3: Εάν βρέθηκε ένας κόμβος στόχος, τότε ανακοίνωσε επιτυχία αλλιώς ανακοίνωσε αποτυχία.

Εκτέλεση:

Στο δοσμένο χώρο αναζήτησης και με αρχικό κόμβο τον s:

- Τρέχων κόμβος ο s. Τα παιδιά του s είναι οι κόμβοι $b(6)$, $c(4)$, $d(5)$, όπου ο αριθμός στην παρένθεση δηλώνει την τιμή της ευριστικής συνάρτησης στον κόμβο αυτό.
- Ο κόμβος c είναι το παιδί με τη μικρότερη τιμή ευριστικής, και η τιμή αυτή είναι μικρότερη από την τιμή της ευριστικής στο s, άρα ορίζεται ως τρέχων κόμβος.
- Τρέχων κόμβος ο c. Το μοναδικό παιδί του c είναι ο κόμβος $h(5)$, με τιμή ευριστικής μεγαλύτερη από αυτή του c, άρα δεν τίθεται ως τρέχων κόμβος.
- Δεν βρέθηκε ο τελικός κόμβος και ο αλγόριθμος σταματάει ανακοινώνοντας αποτυχία.



Ο αλγόριθμος κόλλησε σε σημείο τοπικού ελαχίστου.

Ο αλγόριθμος για τον **A***:

- Συνδυάζει τη λογική του Branch & Bound και την Best First.
- Το μονοπάτι επεκτείνεται με τον καλύτερο από όλους τους κόμβους που βρίσκονται στο μέτωπο αναζήτησης του δένδρου.
- Χρησιμοποιείται η σύνθετη ευριστική συνάρτηση f(k) = g(k) + h(k), όπου:
  g(k) η απόσταση της k από την αρχική κατάσταση, η οποία είναι πραγματική και γνωστή.
  h(k) μία εκτίμηση της απόστασης της k από το στόχο (μέσω μιας ευριστικής συνάρτησης).

Εκτέλεση:

1η Εκτέλεση

Η υλοποίησή του θα πραγματοποιηθεί με την χρήση δύο λιστών, αυτής του μετώπου αναζήτησης και αυτής του κλειστού συνόλου. Συγκεκριμένα, κάθε κατάσταση που βρίσκεται πρώτη στην λίστα θα εξάγεται και αν δεν είναι στο κλειστό σύνολο θα εισάγεται σε αυτό. Στην συνέχεια, θα εισάγονται τα παιδιά της κατάστασης και θα ταξινομούνται με βάση την σύνθετη ευριστική f(k) από το μικρότερο στοιχείο στο μεγαλύτερο. Αν από το μέτωπο αναζήτησης εξέλθει το στοιχείο g, τότε τερματίζει ο αλγόριθμος.

Σημειώσεις:

1) Στο κλειστό σύνολο οι καταστάσεις ταξινομούνται αλφαβητικά. Επίσης, καταστάσεις που έχουν ήδη εισαχθεί σε αυτό δεν θα ξανά εισάγονται.

2) Κατά την ταξινόμηση του μετώπου αναζήτησης (μετά την εισαγωγή ενός νέου μονοπατιού, τυχόν ισότητες στην τιμή της f θα αντιμετωπίζονται αλφαβητικά, ξεκινώντας από το γράμμα της τελευταίας κατάστασης του μονοπατιού και πηγαίνοντας προς την πρώτη.

3) Ο συμβολισμός s(abc)10 αντιστοιχεί σε a → b → c → s με κόστος 10.

Οπότε,

Μέτωπο Αναζήτησης:

s9

c(s)6, d(s)7, b(s)11

d(s)7, b(s)11, h(sc)13

b(s)11, i(sd)11, h(sc)13, h(sd)15

k(sb)9, i(sd)11, h(sc)13, e(sb)14, h(sd)15

i(sd)11, h(sc)13, e(sb)14, h(sd)15, g(sbk)18

h(sc)13, e(sb)14, h(sd)15, j(sdi)15, g(sbk)18

e(sb)14, h(sd)15, j(sdi)15, g(sbk)18, j(sch)18

h(sd)15, j(sdi)15, g(sbe)18, g(sbk)18, j(sch)18

j(sdi)15, g(sbe)18, g(sbk)18, j(sch)18

g(sdij)14, g(sbe)18, g(sbk)18, j(sch)18

Κλειστό Σύνολο:

-

s

c, s

c, d, s

b, c, d, s

b, c, d, k, s

b, c, d, i, k, s

b, c, d, e, h, i, k, s

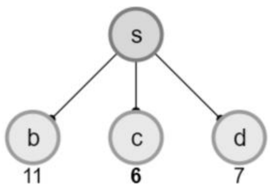
b, c, d, e, h, i, k, s

b, c, d, e, h, i, j, k, s

Συνεπώς, ο αλγόριθμος A* μετέβη στον g με το μονοπάτι: s → d → i → j → g με κόστος 14.

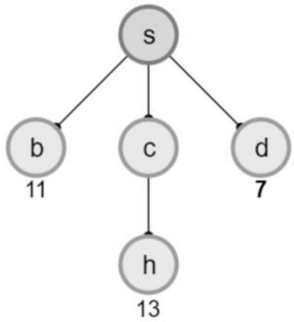
$2^η$ Εκτέλεση:

Εκτελείται στον χώρο αναζήτησης με αρχική κατάσταση την s:



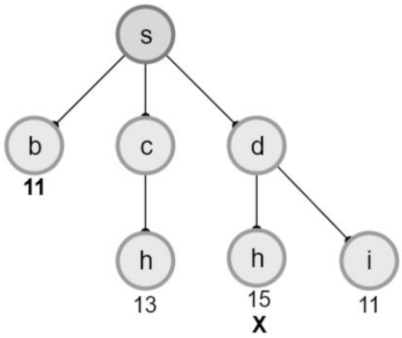
Βήμα 1:

Έπειτα, ανοίγουμε τα παιδιά του s και ταξινομούμε σε αύξουσα σειρά τις ευριστικές. Το μέτωπο αναζήτησης ταξινομημένο είναι οι κόμβοι c, d, b. Μικρότερη τιμή ευριστικής έχει ο c άρα επεκτείνουμε εκεί.



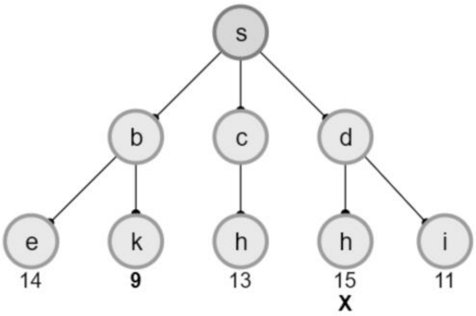
Βήμα 2:

Οι κόμβοι στο μέτωπο αναζήτησης είναι οι d, b, h. Δεν κλαδεύουμε τίποτα. Μικρότερη ευριστική στο d άρα επεκτείνουμε εκεί.



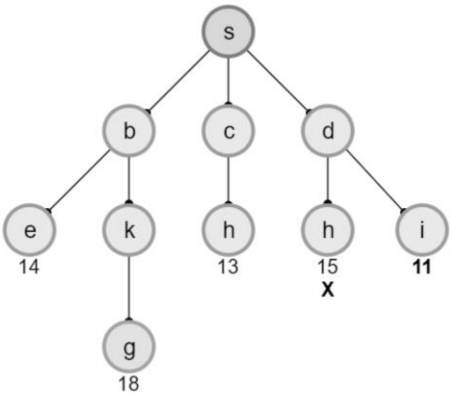
Βήμα 3:

Οι κόμβοι στο μέτωπο αναζήτησης είναι οι b, i, $h(13)$, $h(15)$. Κλαδεύουμε το $h(15)$ γιατί μπορούμε να φτάσουμε στο h με το μονοπάτι $s \rightarrow c \rightarrow h$ με μικρότερη τιμή ευριστικής (13). Συνεχίζουμε επεκτείνοντας στο b (κατά σύμβαση αν δύο κόμβοι έχουν ίση τιμή ευριστικής ταξινομούνται κατ' αλφαβητική σειρά).



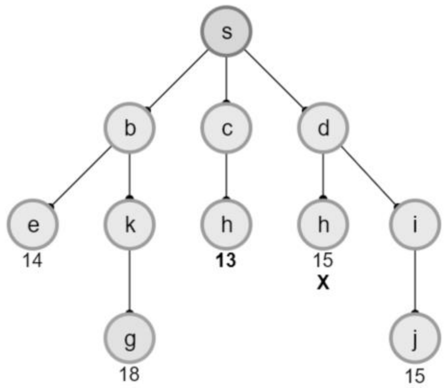
Βήμα 4:

Στο μέτωπο αναζήτησης είναι οι k, i, h, e. Επεκτείνουμε στο k.



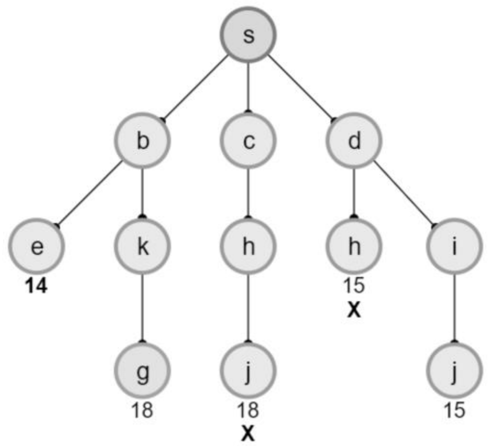
Βήμα 5:

Βρέθηκε πιθανή λύση με κόστος 18 το μονοπάτι $s \rightarrow b \rightarrow k \rightarrow g$. Από εδώ και πέρα θα κλαδεύουμε όλα τα μονοπάτια μήκους μεγαλύτερο από 18 αφού αποκλείεται να οδηγήσουν σε καλύτερη λύση. Μέτωπο αναζήτησης i, h, e. Επεκτείνουμε στο i.



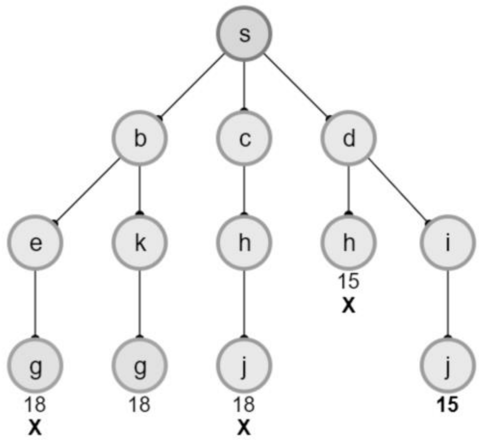
Βήμα 6:

Μέτωπο αναζήτησης h, e, j. Δεν κλαδεύουμε τίποτα. Επεκτείνουμε στο h.



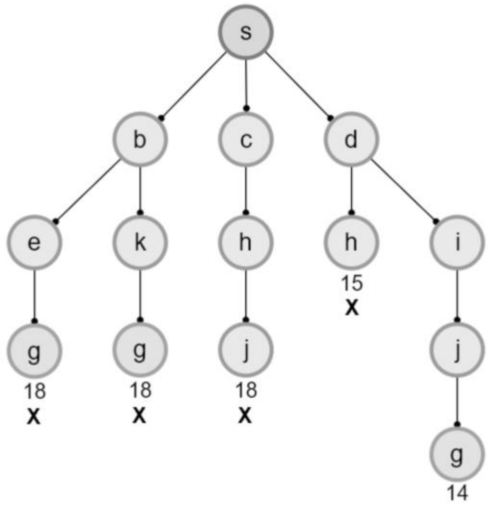
Βήμα 7:

Μέτωπο αναζήτησης e, $j(15)$, $j(18)$. Κλαδεύουμε το μονοπάτι $s \rightarrow c \rightarrow h \rightarrow j(18)$ γιατί μπορούμε να φτάσουμε στο j με μικρότερη τιμή ευριστικής μέσω του μονοπατιού $s \rightarrow d \rightarrow i \rightarrow j(15)$. Επεκτείνουμε το e.



Βήμα 8:

Βρήκαμε πιθανή λύση $s \rightarrow b \rightarrow e \rightarrow g(18)$. Δεν είναι καλύτερη από την βέλτιστη ως τώρα άρα απορρίπτεται. Μέτωπο αναζήτησης είναι μονάχα ο κόμβος j άρα επεκτείνουμε εκεί.



Βήμα 9:

Βρέθηκε η λύση $s \rightarrow d \rightarrow i \rightarrow j \rightarrow g(14)$, καλύτερη από τη βέλτιστη ως τώρα (18) άρα την κρατάμε ως καλύτερη λύση. Το μέτωπο αναζήτησης είναι άδειο άρα ο αλγόριθμος A* τελείωσε.

Βρέθηκε μονοπάτι που οδηγεί στο στόχο άρα ανακοινώνει επιτυχία με βέλτιστη λύση την $s \rightarrow d \rightarrow i \rightarrow j \rightarrow g(14)$.

2. *Πόσες λύσεις έχει το πρόβλημα και ποια είναι η βέλτιστη λύση του προβλήματος; Βρίσκουν τη βέλτιστη λύση οι παραπάνω αλγόριθμοι; Για αυτούς που τη βρίσκουν μπορούμε να είμαστε βέβαιοι εκ των προτέρων ότι θα τη βρουν με βάση τα χαρακτηριστικά του προβλήματος;*

<u>Σύνολο λύσεων</u>:
s → d → h → j → g (με κόστος 19)
s → b → e → g (με κόστος 18)
s → b → k → g (με κόστος 18)
s → c → h → j → g (με κόστος 17)
s → d → i → j → g (με κόστος 14)

Εν προκειμένω, η βέλτιστη λύση βρίσκεται από τον A* και όχι από τον Hill Climbing και είναι:
s → d → i → j → g
Παρατηρούμε ότι ο hill climbing algorithm δε βρήκε τη βέλτιστη λύση, δεν βρήκε καν λύση, επειδή κόλλησε σε τοπικό ελάχιστο. Το συγκεκριμένο είναι σύνηθες πρόβλημα του hill climbing algorithm, και υπάρχουν διάφοροι τρόποι για να αντιμετωπιστεί, όπως η τυχαία επιλογή κόμβου ύστερα από κάθε στιγμή που κολλάει και στη συνέχεια σύνδεση των μονοπατιών (random walks).

Ωστόσο, ο A* δεν είναι βέβαιο ότι θα βρει την βέλτιστη λύση, αν αναλογισθούμε ότι ο ευριστικός μηχανισμός h[n] δεν είναι αποδεκτός. Συγκεκριμένα, ο h[n] λέγεται αποδεκτός αν -για κάθε κατάσταση του χώρου- είναι μικρότερος ή ίσος από την πραγματική απόσταση της n από την τελική κατάσταση, δηλ. $h(n) \leq d(n,g)$ όπου $h(n)$ η τιμή της ευριστικής h στο n και $d(n,g)$ η πραγματική απόσταση του n από το στόχο. Στην περίπτωσή μας όμως ισχύει $h(j) > d(j,g)$ αφού $h(j)=3$ και $d(j,g)=2$ άρα ο ευριστικός μηχανισμός δεν εγγυάται ότι θα βρεθεί η βέλτιστη λύση. Εν προκειμένω, η κατάσταση j έχει h[j] = 3, ενώ η απόσταση j → g έχει πραγματικό κόστος 2. Παρ' όλα αυτά, η τιμή h[j] δεν είναι τέτοια, ώστε να αποτρέψει τον A* από την εύρεση της βέλτιστης λύσης.

<u>2^{η} Άσκηση (Ευφυείς Πράκτορες – Θέμα)</u>

1. *Να καθορίσετε το περιβάλλον, τους αισθητήρες, τις δράσεις και τους δείκτες επίδοσης, κάνοντας τις απαραίτητες αφαιρέσεις.*

   <u>Περιβάλλον</u>: Στο περιβάλλον περιλαμβάνονται τα αντικείμενα που καλείται να πλοηγήσει το σύστημα και τα οποία ανήκουν σε μία από τις 5 διαφορετικές κατηγορίες αντικειμένων. Επιπλέον, έχουμε τυχόν εμπόδια, δηλαδή αμετακίνητα αντικείμενα του αυτοκινητοδρόμου, τα οποία καλείται να αποφεύγει το όχημα. Προφανώς, τα εμπόδια δεν ανήκουν στα αντικείμενα προς πλοήγηση. Τέλος, είναι πιθανή η συνύπαρξη του οχήματος με άλλους ανθρώπους ή/και πράκτορες.

   <u>Αισθητήρες</u>: Όσον αφορά τον προσδιορισμό της δικής του θέσης, το όχημα πρέπει να διαθέτει τους ανάλογους αισθητήρες. Παραδείγματα τέτοιων αισθητήρων είναι το GPS ή/και οι κάμερες χώρου. Όσον αφορά το προσδιορισμό της θέσης άλλων αντικειμένων, είναι αναγκαία η ύπαρξη αισθητήρων υπερήχων και θερμότητας, για την αναγνώριση κοντινών αντικειμένων και ανθρώπων αντίστοιχα.

   <u>Ενέργειες</u>: Ξεκινώντας με την κίνηση του οχήματος, έχουμε την κίνηση προς τα εμπρός και προς τα πίσω, ανάλογα με την επιθυμητή κατεύθυνση κίνησης. Μάλιστα, τυχόν κινήσεις αριστερά, δεξιά ή διαγώνια περιλαμβάνουν την περιστροφή του ρομπότ γύρω από τον εαυτό του, πιθανόν με την χρήση πολικών συντεταγμένων, και στην συνέχεια την εμπρόσθια ή προς τα πίσω μετακίνηση. Σχετικά με τα αντικείμενα του χώρου, το όχημα καλείται να επιτελεί την αποφυγή τους.

   <u>Δείκτες επίδοσης</u>: Από την πλευρά της αξιοπιστίας, κρίνονται απαραίτητοι δείκτες που αφορούν την ασφάλεια του χώρου και τους γύρω ανθρώπους-πράκτορες, καθώς και την ποιότητα κατασκευής του οχήματος (πριν, κατά την διάρκεια και μετά το τέλος των διεργασιών στον αυτοκινητόδρομο). Από την πλευρά της απόδοσης, δείκτες επίδοσης για την ασφάλεια που παρέχει, την κατανάλωση ενέργειας του οχήματος, καθώς και την ταχύτητα του (τόσο ως προς την ταχύτητα υλοποίησης της μίας ενέργειας, όσο και ως προς την βέλτιστη διαδοχή ενεργειών) είναι εξίσου σημαντικά εργαλεία για την βελτιστοποίηση του εγχειρήματος.

2. *Να καθορίσετε τον κόσμο του προβλήματος και να δώσετε ένα παράδειγμα μίας κατάστασης του κόσμου. Να καθορίσετε τους τελεστές μετάβασης από μία κατάσταση σε μία άλλη, και να δώσετε μερικά παραδείγματα τελεστών.*

   Η υλοποίηση του κόσμου του προβλήματος μπορεί να γίνει μέσω μίας σειράς μεταβλητών-ιδιοτήτων, οι οποίες έχουν να κάνουν με το περιβάλλον, τους αισθητήρες και τις ενέργειες του οχήματος (βλέπε ερώτημα (1)). Μάλιστα, κάθε κατηγορία αντικειμένου (ρομπότ, εμπόδια, προορισμός) θα έχει τις δικές του μεταβλητές. Πιο αναλυτικά:

   <u>Όχημα</u>:
   car_name // κωδική ονομασία οχήματος
   car_position(robot_name) // θέση του οχήματος στο δρόμο (*)
   car_velocity(robot_name) // μέτρο ταχύτητας του οχήματος στο δρόμο
   car_acceleration(robot_name) // επιτάχυνση/επιβράδυνση του οχήματος στο δρόμο

Αντικείμενα:
object_name // κωδική ονομασία αντικειμένου
object_category(object_name) // κατηγορία αντικειμένου
current_position(object_name) // τρέχουσα θέση αντικειμένου (*)
wanted_position(object_name) // επιθυμητή θέση αντικειμένου (*)

(*) Οι συντεταγμένες που θα προκύψουν από καθεμία από τις παραπάνω συναρτήσεις θα είναι ως προς σύστημα αξόνων, το οποίο έχει ως αρχή (0,0) την πόρτα του δωματίου.

Εμπόδια:
obstacle_name // κωδική ονομασία εμποδίου
position(obstacle_name) // θέση εμποδίου στο δωμάτιο (*)

Ως παράδειγμα, μπορούμε να ορίσουμε την αρχική κατάσταση, δηλαδή την στιγμή στην οποία το όχημα εισέρχεται στο δρόμο. Ορίζουμε 5 διαφορετικές κατηγορίες αντικειμένων, με ένα αντικείμενο ανά κατηγορία, καθώς και 2 εμπόδια. Έτσι, έχουμε τα εξής:

Όχημα:
car_name = A
car_position(A) = (0,0) // Η είσοδος του δρόμου ως αρχή των αξόνων (X,Y)
car_velocity(A) = 0 m/s
car_acceleration(A) = 0 m/s

Αντικείμενο_1:
object_name = B
object_category(B) = Αντικείμενο_1
current_position(B) = (1,2,0)
wanted_position(B) = (1,2,0)

Αντικείμενο_2:
object_name = C
object_category(C) = Αντικείμενο_2
current_position(C) = (-4,1,2)
wanted_position(C) = (-4,10,2)

Αντικείμενο_3:
object_name = D
object_category(D) = Αντικείμενο_3
current_position(D) = (5, 3, 5)
wanted_position(D) = (-2, 7, 1)

Αντικείμενο_4:
object_name = E
object_category(E) = Αντικείμενο_4
current_position(E) = (1,1,1)
wanted_position(E) = (1,1,1)

Αντικείμενο_5:
object_name = F
object_category(F) = Αντικείμενο_5
current_position(F) = (6, 6, 4)
wanted_position(F) = (6, 1, 4)

Εμπόδιο_1:
obstacle_name = G
position(G) = (0, 1, 0)

Εμπόδιο_ 2:
obstacle_name = F
position(G) = (0, 0, 20)

Με βάση τις ιδιότητες των αντικειμένων και τα ζητούμενα του προβλήματος, προκύπτουν οι κάτωθι σχέσεις:
O A χρειάζεται να μεταβεί στα αντικείμενα C,D,E,F
O A χρειάζεται να προσέξει τα αντικείμενα D,E
O A χρειάζεται να προσέξει τα αντικείμενα C,D,F
O A χρειάζεται να προσέξει -αν χρειαστεί- το εμπόδιο G
O A δεν χρειάζεται να ασχοληθεί με το αντικείμενο B // current_position(B)=wanted_position(B)
O A δεν χρειάζεται να ασχοληθεί με το εμπόδιο F // position_z_G > position_z_I, με I= {B,C,D,E}

Ως τελεστής μετάβασης ορίζεται η αντιστοίχιση μίας κατάστασης του κόσμου σε μία άλλη. Η μετάβαση αυτή πραγματοποιείται με μία σειρά από ενέργειες μετάβασης (έστω e) και κάθε ενέργεια μετάβασης αντιστοιχεί σε ένα κόστος μετάβασης (έστω c). Στις ενέργειες μετάβαση όσον αφορά το παράδειγμα του ρομπότ- περιλαμβάνονται όλες οι ενέργειες που μπορεί να πραγματοποιήσει το ρομπότ και η υλοποίηση της καθεμίας θα έχει ανάλογο κόστος. Έτσι, έχουμε τα εξής ζεύγη e-c:

Κίνηση οχήματος:
Κίνηση μπροστά: e1-c1
Κίνηση πίσω: e2-c2
Περιστροφή: e3-c3
Επιτάχυνση-Επιβράδυνση: e4-c4

Αλληλεπίδραση με αντικείμενα:
Προσπέραση αντικειμένου: e5-c5
Αποφυγή αντικειμένου: e6-c6

3. *Να σχεδιάσετε ευριστικές συναρτήσεις που εκτιμούν τόσο το κόστος μετάβασης από μία κατάσταση σε μία άλλη, όσο και το υπολειπόμενο κόστος μέχρι την τελική κατάσταση.*

   Θεωρούμε ότι η μετάβαση από την μία κατάσταση στην άλλη περιλαμβάνει το σύνολο ενεργειών, έτσι ώστε να προσπεράσει και να αποφύγει ένα αντικείμενο. Συνεπώς, η νέα κατάσταση θα περιλαμβάνει μόνο ένα επιπλέον αντικείμενο. Μία ενδεικτική ευριστική συνάρτηση είναι η εξής:

   h1[n] = A*[e1c1+e2c2+e3c3+e4c4+e5c5+e6c6]. Πιο αναλυτικά:

   Οι μεταβλητές e1 ως και e4 αντιστοιχούν σε boolean μεταβλητές οι οποίες παίρνουν την τιμή 1 αν η ενέργεια μετάβασης είναι αναγκαία τουλάχιστον 1 φορά. Μάλιστα, για τη μεταβλητή e6 ισχύει ότι:
   e6=0, if current-wanted == 0
   e6=1, if current-wanted <> 0
   Ο ορός (e1c1+e2c2+e3c3+e4c4) έχει να κάνει με τις κινήσεις που απαιτείται να κάνει το όχημα για την μετάβαση από την μία κατάσταση στην άλλη.
   Ο όρος e6c6 έχει να κάνει με το αν είναι αναγκαία η προσπέραση ή αποφυγή του αντικειμένου ή όχι.
   Ως A ορίζουμε την ευκλείδεια απόσταση (θα την ονομάζουμε πλέον D) ανάμεσα στην τρέχουσα και στην επιθυμητή κατάσταση του αντικειμένου ή αλλιώς
   A=D(object_current_position, object_wanted_position)=D1

   Θεωρώντας την προσπέραση ή αποφυγή ως την πιο δαπανηρή διαδικασία και την εμπρόσθια-πισινή κίνηση την πιο φθηνή, μπορούμε να δώσουμε ενδεικτικές τιμές στα κόστη c:
   c1=c2=1, c3=c4=2, c5=c6=3

   Έτσι η h1[n] γίνεται: h1[e1,…,e6] = D1*[(e1+e2) + 2(e3+e4) + 3(e5+e6)]

   Για την εύρεση της ευριστικής που αφορά το υπολειπόμενο κόστος (από την τρέχουσα κατάσταση στην τελική), μία προσέγγιση είναι η εξής:

   h2[n] = Σ { ((c5e5)I + (c6e6)I))*D1I ή h2[n]= 2 Σ { ((e5)I + (c6)I))*D1I }
   (I=κάθε αντικείμενο προς προσπέραση ή αποφυγή)

   Παρατηρούμε ότι και οι 2 παραπάνω ευριστικές συναρτήσεις έχουν σχεδιασθεί, ώστε να δίνουν παραπάνω έμφαση στην Ευκλείδεια απόσταση ανάμεσα στην τρέχουσα και στην επιθυμητή θέση. Επιπρόσθετα, ο πιο πολύπλοκος όρος προς υπολογισμό είναι ο D1 (αφού D1 = {(x1-x2)^2 + (y1-y2)^2 + (z1-z2)2}$^{1/2}$), με τους υπόλοιπους όρους e1,…,e6 να αποτελούν boolean, συνεπώς εύκολους προς υπολογισμό. Έτσι, τόσο η h1[n], όσο και η h2[n] είναι γρήγορες στον υπολογισμό και κατατοπιστικές για το ζητούμενο κόστος.

$3^η$ Άσκηση (Ταξινομητές Naive Bayes)

Σύνολο μηνυμάτων που	Είναι διαφημιστικά	Δεν είναι διαφημιστικά
Περιέχουν τη λέξη «ευκαιρία»	285	225
Δεν περιέχουν τη λέξη «ευκαιρία»	15	1275
Περιέχουν τη λέξη «μοναδικός»	240	600
Δεν περιέχουν τη λέξη «μοναδικός»	60	900

Σύνολο 1800 μηνύματα, που χωρίζονται σε δύο κλάσεις: «είναι διαφημιστικό», «όχι διαφημιστικό».

Έχουμε δύο πιθανά χαρακτηριστικά και τα συμπληρωματικά τους: «ευκαιρία», «όχι ευκαιρία», «μοναδικός», «όχι μοναδικός».

Θεωρώ ότι η ύπαρξη «μοναδικός» και η ύπαρξη «ευκαιρία» είναι ανεξάρτητα (απαραίτητη υπόθεση για να λειτουργήσει ο ταξινομητής.

Υπολογίζουμε τις πιθανότητες:

p(είναι διαφημιστικό) = 300/1800

p(όχι διαφημιστικό) = 1500/1800

p(ευκαιρία | διαφημιστικό) = 285/300

p(όχι ευκαιρία | διαφημιστικό) = 15/300

p(μοναδικός | διαφημιστικό) = 240/300

p(όχι μοναδικός | διαφημιστικό) = 60/300

p(ευκαιρία | όχι διαφημιστικό) = 225/1500

p(όχι ευκαιρία | όχι διαφημιστικό) = 1275/1500

p(μοναδικός | όχι διαφημιστικό) = 600/1500

p(όχι μοναδικός | όχι διαφημιστικό) = 900/1500

Ο τύπος του Bayes:

$$p(i \mid x) = \frac{p(i) \cdot \prod_{k=1}^{p} p(x^{(k)} \mid i)}{\sum_{j=1}^{c} p(j) \cdot \prod_{k=1}^{p} p(x^{(k)} \mid j)}$$

Χρησιμοποιώντας τον τύπο αυτό, όταν μας βρεθεί μήνυμα με 2 χαρακτηριστικά («ευκαιρία» ή «όχι ευκαιρία», «μοναδικός» ή «όχι μοναδικός»), υπολογίζουμε την πιθανότητα να είναι «διαφημιστικό» και την πιθανότητα να είναι «όχι διαφημιστικό», δεδομένου των χαρακτηριστικών αυτών. Έπειτα τον ταξινομούμε ανάλογα με την μεγαλύτερη πιθανότητα.

Συγκεκριμένα, για ένα μήνυμα που περιέχει τη λέξη «ευκαιρία» και τη λέξη «μοναδικός»:

p(διαφημιστικό | ευκαιρία, μοναδικός) =

$$= \frac{p(\delta\iota\alpha\varphi)p(\varepsilon\upsilon\kappa\alpha\iota\rho\acute{\iota}\alpha|\delta\iota\alpha\varphi)p(\mu o\nu\alpha\delta\iota\kappa\acute{o}\varsigma|\delta\iota\alpha\varphi)}{p(\delta\iota\alpha\varphi)p(\varepsilon\upsilon\kappa\alpha\iota\rho\acute{\iota}\alpha|\delta\iota\alpha\varphi)p(\mu o\nu\alpha\delta\iota\kappa\acute{o}\varsigma|\delta\iota\alpha\varphi)+p(o\chi\iota\ \delta\iota\alpha\varphi)p(\varepsilon\upsilon\kappa\alpha\iota\rho\acute{\iota}\alpha|o\chi\iota\ \delta\iota\alpha\varphi)p(\mu o\nu\alpha\delta\iota\kappa\acute{o}\varsigma|o\chi\iota\ \delta\iota\alpha\varphi)}$$

$$= \frac{\frac{300}{1800}*\frac{285}{300}*\frac{240}{300}}{\frac{300}{1800}*\frac{285}{300}*\frac{240}{300}+\frac{1500}{1800}*\frac{225}{1500}*\frac{600}{1500}} = \frac{0{,}13}{0{,}13+0{,}05} = 0{,}72$$

p(όχι διαφημιστικό | ευκαιρία, μοναδικός) =

$$= \frac{p(o\chi\iota\ \delta\iota\alpha\varphi)p(\varepsilon\upsilon\kappa\alpha\iota\rho\acute{\iota}\alpha|o\chi\iota\ \delta\iota\alpha\varphi)p(\mu o\nu\alpha\delta\iota\kappa\acute{o}\varsigma|o\chi\iota\ \delta\iota\alpha\varphi)}{p(\delta\iota\alpha\varphi)p(\varepsilon\upsilon\kappa\alpha\iota\rho\acute{\iota}\alpha|\delta\iota\alpha\varphi)p(\mu o\nu\alpha\delta\iota\kappa\acute{o}\varsigma|\delta\iota\alpha\varphi)+p(o\chi\iota\ \delta\iota\alpha\varphi)p(\varepsilon\upsilon\kappa\alpha\iota\rho\acute{\iota}\alpha|o\chi\iota\ \delta\iota\alpha\varphi)p(\mu o\nu\alpha\delta\iota\kappa\acute{o}\varsigma|o\chi\iota\ \delta\iota\alpha\varphi)}$$

$$= \frac{\frac{1500}{1800}*\frac{225}{1500}*\frac{600}{1500}}{\frac{300}{1800}*\frac{285}{300}*\frac{240}{300}+\frac{1500}{1800}*\frac{225}{1500}*\frac{600}{1500}} = \frac{0{,}05}{0{,}13+0{,}05} = 0{,}28$$

Άρα ένας μήνυμα που γράφει «ευκαιρία» και «μοναδικός» ταξινομείται ότι ως «διαφημιστικό».

4η Άσκηση (Entity Relationship Model – Μοντέλο Οντοτήτων Συσχετίσεων)