

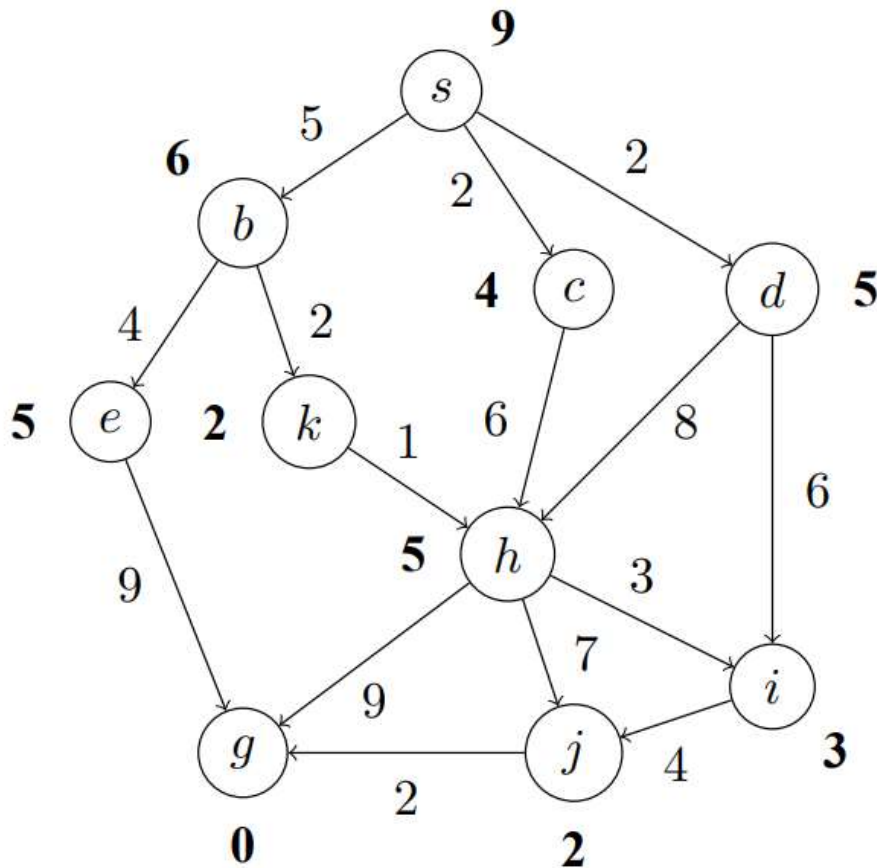
Θεμελιώδη Θέματα Επιστήμης Υπολογιστών

3^η Σειρά Γραπτών Ασκήσεων

(τεχνητή νοημοσύνη)

Άσκηση 1.

1.



Αρχικά, θα εκτελέσουμε τον αλγόριθμο αναρρίχησης λόφου.

ΒΗΜΑ	ΜΕΤΩΠΟ ΑΝΑΖΗΤΗΣΗΣ	ΚΛΕΙΣΤΟ ΣΥΝΟΛΟ
1	[s]	\emptyset
2	[c]	[s]

Ο αλγόριθμος καταλήγει σε αποτυχία.

Αναλυτικότερα, στο πρώτο βήμα ορίζουμε τον αρχικό κόμβο [s], ενώ το κλειστό σύνολο ισούται με το κενό, εφόσον δεν υπάρχουν ακόμη visited nodes. Στο δεύτερο βήμα εξετάζουμε τον κόμβο με την μικρότερη τιμή της ευριστικής εκτίμησης της απόστασης μέχρι την τελική κατάσταση, δηλαδή τον κόμβο c. Το κλειστό σύνολο αποτελείται από τον μόνο visited κόμβο, δηλαδή τον [s].

Προσπαθώντας να εκτελέσουμε το τρίτο βήμα, παρατηρούμε πως ο μόνος προσβάσιμος κόμβος από τον κόμβο c, είναι ο κόμβος h. Ωστόσο, ο δεύτερος έχει μεγαλύτερη τιμή ευριστικής απόστασης

από τον πρώτο, επομένως (σύμφωνα με τη θεωρία του αλγορίθμου Climbing Hill), ο αλγόριθμος τερματίζει με αποτυχία.

Στη συνέχεια, θα εκτελέσουμε τον αλγόριθμο A*.

ΒΗΜΑ	ΛΙΣΤΑ ΜΟΝΟΠΑΤΙΩΝ
1	[s] (ορίζουμε τον αρχικό κόμβο)
2	[s → b 11, s → c 6, s → d 7]
3	[s → b 11, s → d 7, s → c → h 13]
4	[s → b 11, s → c → h 13, s → d → h 15, s → d → i 11] Παρατηρούμε πως υπάρχουν δύο μονοπάτια για τη διαδρομή [s → h], επομένως διαγράφουμε από τη λίστα μονοπατιών το μονοπάτι [s → h] με το μεγαλύτερο κόστος διαδρομής. Συνεπώς: [s → b 11, s → c → h 13, s → d → i 11]
5	[s → b 11, s → c → h 13, s → d → i → j 14]
6	[s → c → h 13, s → d → i → j 14, s → b → e 14, s → b → k 9]
7	[s → c → h 13, s → d → i → j 14, s → b → e 14, s → b → k → h 13] Όμοια με το βήμα 4, διαθέτουμε δύο μονοπάτια για τη διαδρομή [s → h], επομένως θα διαγράψουμε αυτό με το μεγαλύτερο κόστος διαδρομής (τυγχάνει να έχουν ίδιο κόστος, οπότε δεν έχει σημασία ποιο από τα δύο διαγράφεται): [s → c → h 13, s → d → i → j 14, s → b → e 14]
8	[s → d → i → j 14, s → b → e 14, s → c → h → i 14, s → c → h → j 16, <u>s → c → h → g 17</u>] Σε αυτό το βήμα έχουμε μερική επιτυχία του αλγορίθμου, καθώς εντοπίζει ένα μονοπάτι για τη διαδρομή [s → g], όμως δεν γνωρίζουμε ακόμη αν αυτό είναι το βέλτιστο. Επίσης, όμοια με τα βήματα 4 και 7, διαθέτουμε δύο μονοπάτια για τη διαδρομή [s → j], οπότε διαγράφουμε αυτό με το μεγαλύτερο κόστος διαδρομής: [s → c → h → g 17, s → d → i → j 14, s → b → e 14, s → c → h → i 14]
9	[s → c → h → g 17, s → d → i → j 14, s → b → e 14, s → c → h → i → j 17] Και πάλι, διαθέτουμε δύο μονοπάτια για τη διαδρομή [s → j], οπότε διαγράφουμε αυτό με το μεγαλύτερο κόστος: [s → c → h → g 17, s → d → i → j 14, s → b → e 14]
10	[s → c → h → g 17, s → b → e 14, <u>s → d → i → j → g 14</u>] Όμοια με το βήμα 8, έχουμε και πάλι μερική επιτυχία, εφόσον εντοπίστηκε ένα μονοπάτι για τη διαδρομή [s → g]. Σε αυτό το βήμα, διαγράφουμε όλες τις μη-ολοκληρωμένες διαδρομές με κόστος διαδρομής μεγαλύτερο ή ίσο με το μικρότερο κόστος ολοκληρωμένης διαδρομής, δηλαδή με κόστος ≥ 14 .
11	[s → c → h → g 17, <u>s → d → i → j → g 14</u>]
12	[s → d → i → j → g 14]

Ο αλγόριθμος καταλήγει σε επιτυχία, παραδίδοντας το βέλτιστο μονοπάτι για τη διαδρομή [s → g].

2. Σύμφωνα με όσα ειπώθηκαν στο ερώτημα (1), ο αλγόριθμος αναρρίχησης λόφου καταλήγει σε αποτυχία, ενώ ο αλγόριθμος A^* καταλήγει σε επιτυχία, δίνοντας μάλιστα τη βέλτιστη λύση. Το τελευταίο, ωστόσο, ήταν αναμενόμενο χάρη στη φύση του αλγορίθμου. Πιο συγκεκριμένα, κάθε φορά που βρίσκει ένα μονοπάτι της μορφής [αρχικός κόμβος \rightarrow τελικός κόμβος] με κόστος x , διαγράφει κάθε μη-ολοκληρωμένο μονοπάτι με κόστος $\geq x$. Επίσης, όταν εντοπίζονται μονοπάτια που καταλήγουν σε sink node αυτά επίσης διαγράφονται, ανεξαρτήτως του κόστους διαδρομής τους. Έτσι, σε κάποιο βήμα του αλγορίθμου, η λίστα μονοπατιών θα περιέχει αποκλειστικά μονοπάτια της μορφής [αρχικός κόμβος \rightarrow τελικός κόμβος] με κόστος x_i ($i = 1, \dots, n$) για κάθε i -οστό μονοπάτι. Στο τελευταίο βήμα του αλγορίθμου, παραδίδεται ως λύση το μονοπάτι με το μικρότερο κόστος διαδρομής (χάρη στην ιδιότητα που επιτάσσει πως όταν υπάρχουν παραπάνω από ένα μονοπάτια για μια διαδρομή, διατηρείται η διαδρομή με το μικρότερο κόστος). Συνεπώς, ο αλγόριθμος A^* πάντοτε δίνει βέλτιστη λύση.

Άσκηση 2.

1. Θα καθορίσουμε τα απαραίτητα στοιχεία για το πρόβλημα.

- Περιβάλλον: Το ρομπότ επιλέγει κάθε φορά ένα υλικό, το οποίο ανήκει σε μια από τις 5 κατηγορίες υλικών. Καθώς κινείται στην αποθήκη έχει «επίγνωση» της θέσης κάθε υλικού, ωστόσο στο περιβάλλον του περιλαμβάνονται και τυχόν εμπόδια – τα οποία, προφανώς, πρέπει να αποφεύγει. Τέλος, είναι πιθανή η επαφή του ρομπότ με ανθρώπους.
- Αισθητήρες: Το ρομπότ πρέπει να είναι σε θέση να «αντιλαμβάνεται» τόσο τη δική του θέση στην αποθήκη, όσο και τη θέση των υλικών σε αυτή, αλλά και να αναγνωρίζει τυχόν ανθρώπους με τους οποίους έρχεται σε επαφή. Για τον προσανατολισμό του στον χώρο, υπάρχουν κατάλληλοι αισθητήρες, όπως οι κάμερες χώρου. Για τον προσδιορισμό της θέσης των υλικών αλλά και την αναγνώριση ανθρώπων είναι απαραίτητος ο εξοπλισμός του ρομπότ με αισθητήρες υπερήχων και θερμότητας. Προκειμένου, επίσης, να συγκεντρώνει τα υλικά από τα ράφια, το ρομπότ πρέπει να διαθέτει κατάλληλους αισθητήρες, οι οποίοι θα υπολογίζουν το πλάτος, το μήκος και το βάρος των υλικών, καθώς και την ελάχιστη δύναμη που πρέπει να ασκηθεί προκειμένου να ληφθούν τα υλικά (ειδικά αν γίνεται λόγος για υγρά ή γενικότερα για υλικά που απαιτούν λεπτό χειρισμό). Τέλος, είναι απαραίτητο το ρομπότ να μπορεί να καταμετρά τα υλικά κάθε κατηγορίας, ώστε να αποφαινεται αν είναι δυνατόν να εκτελέσει τη ζητούμενη παραγγελία (αν π.χ. υπάρχουν 5 τεμάχια ενός υλικού και η παραγγελία απαιτεί 6, δεν είναι δυνατή η ολοκλήρωσή της).
- Ενέργειες: Το ρομπότ θα πρέπει να δύναται να κινηθεί προς όλες τις κατευθύνσεις, μπροστά, πίσω, δεξιά, αριστερά, ακόμη και διαγώνια. Εξίσου σημαντική είναι η δυνατότητα περιστροφής του (π.χ. με χρήση πολικών συντεταγμένων). Θα πρέπει η ευθύγραμμη κίνηση να μπορεί να προηγείται και να ακολουθεί μια πιθανή περιστροφή. Όσον αφορά τα υλικά, το ρομπότ ευθύνεται τόσο για την επιλογή τους, όσο και για τη μετακίνησή τους. Πιο συγκεκριμένα, πρέπει να αναγνωρίζει τους 5 διαφορετικούς τύπους υλικών, να μπορεί να «πιάνει» το εκάστοτε υλικό (σύμφωνα με το μέγεθος, το βάρος και την ευαισθησία του, οι οποίες προσδιορίζονται χάρη στους αισθητήρες) και να το μετακινεί προς την επιθυμητή τοποθεσία. Επίσης, είναι απαραίτητη η αναγνώριση τόσο των εμποδίων όσο και των ανθρώπων που υπάρχουν στον χώρο, έτσι ώστε το ρομπότ να τα αποφεύγει.
- Δείκτες επίδοσης: Όσον αφορά στην αξιοπιστία, απαιτούνται δείκτες επίδοσης που διασφαλίζουν τόσο την ασφάλεια της αποθήκης στην οποία κινείται το ρομπότ, όσο και των

ανθρώπων με τους οποίους έρχεται σε επαφή. Κάτι τέτοιο αξιολογείται σε τρία στάδια: πριν, κατά τη διάρκεια και αφού το ρομπότ εκτελέσει τις παραπάνω διεργασίες στον χώρο. Όσον αφορά στην απόδοση, είναι αναγκαίοι δείκτες επίδοσης που αποφαίνονται την αποτελεσματικότητα εκλογής των υλικών, την κατανάλωση ενέργειας του ρομπότ, την αποδοτικότητα του ρομπότ (ρυθμός και σειρά εκτέλεσης εντολών).

2. Μπορούμε να κατασκευάσουμε τον κόσμο του προβλήματος κάνοντας χρήση ψευδοκώδικα, για κάθε μια από τις διαφορετικές κατηγορίες που περιγράφηκαν αναλυτικά στο ερώτημα (1).

- Ρομπότ:
robot_name
robot_position(robot_name) – αλγόριθμος για την εύρεση θέσης του ρομπότ στον χώρο
robot_speed(robot_name)
Θα θεωρήσουμε πως η θέση του ρομπότ στον χώρο προσδιορίζεται βάσει ενός συστήματος καρτεσιανών συντεταγμένων με καθορισμένο το σημείο $O(0, 0)$ έτσι ώστε να ταυτίζεται, π.χ. με την πόρτα.
- Υλικά:
ingredient_name – όνομα υλικού
ing_category(ingredient_name) – κατηγορία στην οποία ανήκει το υλικό
ing_quantity(ingredient_name) – αριθμός τεμαχίων που ζητούνται
ing_position_initial(ingredient_name) – πού βρίσκεται το υλικό
ing_position_eventual(ingredient_name) – πού πρέπει να βρεθεί το υλικό
ing_info(ingredient_name) – περιλαμβάνει τα στοιχεία του υλικού που αφορούν στο μέγεθος, στο βάρος του και στο πόσο εύθραυστο είναι
ing_status(ingredient_name) – αποφαινεται κατά πόσο υπάρχει επαρκής ποσότητα από το ζητούμενο υλικό
- Εμπόδια/Άνθρωποι:
obstacle_name
obstacle_type(obstacle_name) – καθορίζει αν πρόκειται για αντικείμενο ή για άνθρωπο
obstacle_position(obstacle_name)

Θα εφαρμόσουμε τον παραπάνω ψευδοκώδικα και για τις 5 διαφορετικές κατηγορίες υλικών, καθώς και για τις 2 κατηγορίες εμποδίων (αντικείμενο/άνθρωπος). Ακολουθεί τυχαίο παράδειγμα:

- Ρομπότ:
Clarence
robot_position(Clarence) = (0, 0) – αρχικοποίηση με το ρομπότ να βρίσκεται στο $O(0, 0)$
robot_speed(Clarence) = 0 m/s
- Υλικά:
 - Υλικά Κατηγορίας 1:
bread
ing_category(bread) = 1 – υλικό κατηγορίας 1

ing_quantity(bread) = 1
ing_position_initial(bread) = (0, 1, 2)
ing_position_eventual(bread) = (4, 2, 2)
ing_info(bread) = 10 cm (width), 400 grams (weight), not sensitive
ing_status(bread) = available

○ Υλικά Κατηγορίας 2:

cheese

ing_category(cheese) = 2 – υλικό κατηγορίας 2

ing_quantity(cheese) = 3

ing_position_initial(cheese) = (4, 1, 5) – και για τα 3 τεμάχια

ing_position_eventual(cheese) = (4, 1, 5)

ing_info(cheese) = 20 cm (width), 1 kg (weight), not sensitive

ing_status(cheese) = available

○ Υλικά Κατηγορίας 3:

cake

ing_category(cake) = 3 – υλικό κατηγορίας 3

ing_quantity(cake) = 5

ing_position_initial(cake) = (0, 5, 2)

ing_position_eventual(cake) = (-2, 8, 1)

ing_info(cake) = 85 cm (width), 2 kg (weight), sensitive

ing_status(cake) = unavailable

○ Υλικά Κατηγορίας 4:

apple

ing_category(apple) = 4 – υλικό κατηγορίας 4

ing_quantity(apple) = 2

ing_position_initial(apple) = (-1, 2, 1)

ing_position_eventual(apple) = (0, 0, 1)

ing_info(apple) = 5 cm (width), 20 grams (weight), not sensitive

ing_status(apple) = available

○ Υλικά Κατηγορίας 5:

cookies

ing_category(cookies) = 5 – υλικό κατηγορίας 5

ing_quantity(cookies) = 6

ing_position_initial(cookies) = (5, 1, 2)

ing_position_eventual(cookies) = (5, 1, 2)

ing_info(cookies) = 15 cm (width), 10 grams (weight), sensitive

ing_status(cookies) = available

• Εμπόδια:

○ Εμπόδιο Κατηγορίας 1:

chair

obstacle_type(chair) = object – πρόκειται για αντικείμενο

obstacle_position(chair) = (0, 0, 6)

ο Εμπόδιο Κατηγορίας 2:

Emma

obstacle_type(Emma) = human – πρόκειται για άνθρωπο

obstacle_position(Emma) = (1, 2, 1)

Σύμφωνα με όσα έχουμε ορίσει παραπάνω, έχουμε το εξής αποτέλεσμα:

Ο Clarence θα πρέπει να μετακινήσει τα υλικά bread και apple. Το υλικό cake δεν είναι διαθέσιμο, καθώς η ζητούμενη ποσότητα είναι μεγαλύτερη από την υπάρχουσα ποσότητα, επομένως ο Clarence θα εμφανίσει κάποιο μήνυμα αποτυχίας. Με τα υπόλοιπα υλικά (cheese, cookies) δεν είναι απαραίτητο να ασχοληθεί, καθώς για καθένα από αυτά ισχύει ing_status=moved.

Ο Clarence δεν πρέπει να «προσέξει» τη δύναμη με την οποία θα πιάσει τα υλικά bread και apple (ing_info_sensitivity = not sensitive). Αντίθετα, πρέπει να ασκήσει λιγότερη δύναμη στο υλικό cake (ing_info_sensitivity = sensitive) – σε περίπτωση που υπάρχει η απαιτούμενη ποσότητα τεμαχίων.

Ο Clarence ίσως χρειαστεί να αποφύγει το εμπόδιο Emma. Ωστόσο, δεν χρειάζεται να ανησυχούμε για το εμπόδιο chair, καθώς position_chair_z = 6 > position_ingredient_z (όπου ingredient ένα από τα παραπάνω υλικά).

-Τέλος Παραδείγματος-

Στη συνέχεια θα ασχοληθούμε με τη μετάβαση κατάστασης. Μια μετάβαση κατάστασης συνδέει δύο καταστάσεις μέσω μιας ενέργειας E, η οποία έχει κόστος K. Ορίζοντας, για το παραπάνω παράδειγμα, μεταβάσεις κατάστασης, καταλήγουμε στο εξής:

- Μετακίνηση στον χώρο:
Μετακίνηση μπροστά: E1-K1
Μετακίνηση πίσω: E2-K2
Μετακίνηση δεξιά: E3-K3
Μετακίνηση αριστερά: E4-K4
Περιστροφή: E5-K5
- Αλληλεπίδραση με αντικείμενα:
Πιάσιμο αντικειμένου: E6-K6
Μεταφορά αντικειμένου: E7-K7

3. Θεωρούμε ότι η μετάβαση από τη μία κατάσταση στην άλλη περιλαμβάνει το σύνολο ενεργειών, έτσι ώστε να πιαστεί και να μεταφερθεί το πολύ ένα αντικείμενο. Άρα, η προκύπτουσα κατάσταση περιλαμβάνει μόνο ένα επιπλέον moved αντικείμενο. Μια ενδεικτική ευριστική συνάρτηση φαίνεται παρακάτω:

$$h1[n] = \text{distance} * (E1K1 + E2K2 + E3K3 + E4K4 + E5K5 + E6K6 + E7K7)$$

Αναλυτικότερα, οι μπορούμε να αντιστοιχήσουμε τις μεταβλητές E1, E2, E3, E4 και E5 σε Boolean μεταβλητές, οι οποίες ισούνται με 1 αν η εκάστοτε κίνηση του ρομπότ πραγματοποιείται τουλάχιστον μια φορά. Αντίστοιχα, για τις μεταβλητές E6 και E7 θεωρούμε ότι :

```
if (ing_status == moved) {
    E6 = 0
    E7 = 0
}
Else {
    E6 = 1
    E7 = 1
}
```

Έτσι, για την ενδεικτική ευριστική συνάρτηση h1 μπορούμε να διακρίνουμε διαφορετικούς όρους:

- Ο όρος (E1K1 + E2K2 + E3K3 + E4K4 + E5K5) αντιστοιχεί στις κινήσεις που πραγματοποιεί το ρομπότ κατά τη μετάβαση μεταξύ καταστάσεων, ενώ
- Ο όρος (E6K6 + E7K7) αντιστοιχεί στην αναγκαιότητα –ή μη- του πιάσιμου και της μετακίνησης του υλικού.
- Ο όρος distance αντιστοιχεί στη διαφορά (ing_position_initial – ing_position_eventual).

Αν θεωρήσουμε πως το πιάσιμο και η μεταφορά κάθε υλικού είναι η πιο δαπανηρή διαδικασία, ενώ η γραμμική κίνηση του ρομπότ είναι η πιο φθηνή, μπορούμε να πούμε πως K1=K2=K3=K4=1, K5=2, K6=K7=3, άρα, αντικαθιστώντας, η h1 γίνεται:

$$h1[n] = distance * [1 * (E1 + E2 + E3 + E4) + 2 * E5 + 3 * (E6 + E7)]$$

Όσον αφορά την ευριστική συνάρτηση για το υπολοιπόμενο κόστος (κατά τη μετάβαση από μια κατάσταση σε μια άλλη) μπορούμε να πούμε το εξής:

$$h2[n] = \sum \{ (E6K6 * ingredient + E7K7 * ingredient) * distance \} \text{ (όπου ingredient το εκάστοτε υλικό)}$$

$$h2[n] = \sum \{ 2 * (E6 + E7) * ingredient * distance \}$$

Τόσο η h1[n] όσο και η h2[n] δίνουν έμφαση στην Ευκλείδεια απόσταση ανάμεσα στην αρχική και την τελική θέση. Η απόσταση αυτή είναι ο πιο πολύπλοκος όρος των συναρτήσεων και έχει τύπο:

$$distance = [(x_eventual - x_initial)^2 + (y_eventual - y_initial)^2 + (z_eventual - z_initial)^2] * \frac{1}{2}$$

Άσκηση 3.

Έχουμε τα εξής στοιχεία για τους λαμπτήρες που έχει λάβει ο ιδιοκτήτης του καταστήματος:

Πίνακας 1: Πίνακας προμήθειας ηλεκτρολογικού υλικού

Σύνολο λαμπτήρων που	προήλθαν από τον Α	προήλθαν από τον Β	προήλθαν από τον Γ
είναι ελαττωματικοί	15	25	40
δεν είναι ελαττωματικοί	85	175	260
είναι αλογόνου	50	120	200
δεν είναι αλογόνου	50	80	100

Διαθέτουμε, λοιπόν, συνολικά 600 λαμπτήρες, οι οποίοι χωρίζονται σε τρεις κλάσεις, ανάλογα με το αν προέρχονται από τον κατασκευαστή Α (ορίζω ενδεχόμενο Α), αν προέρχονται από τον κατασκευαστή Β (ορίζω ενδεχόμενο Β) και αν προέρχονται από τον κατασκευαστή Γ (ορίζω ενδεχόμενο Γ). Επίσης διακρίνονται σε κατηγορίες ανάλογα με τα χαρακτηριστικά που έχουν, δηλαδή αν είναι ελαττωματικοί (ορίζω ενδεχόμενο Ε) ή όχι (Ε') και ανάλογα με το αν είναι αλογόνου (ορίζω ενδεχόμενο Λ) ή όχι (Λ'). Υπολογίζουμε τις πιθανότητες:

$$P(A) = \frac{100}{600} = \frac{1}{6}, P(B) = \frac{200}{600} = \frac{1}{3}, P(\Gamma) = \frac{300}{600} = \frac{1}{2}$$

$$P(A') = \frac{500}{600} = \frac{5}{6}, P(B') = \frac{400}{600} = \frac{2}{3}, P(\Gamma') = \frac{300}{600} = \frac{1}{2}$$

$$P(E|A) = \frac{15}{100} = \frac{3}{20}, P(E|B) = \frac{25}{200} = \frac{1}{8}, P(E|\Gamma) = \frac{40}{300} = \frac{2}{15}$$

$$P(E'|A) = \frac{85}{100} = \frac{17}{20}, P(E'|B) = \frac{175}{200} = \frac{7}{8}, P(E'|\Gamma) = \frac{260}{300} = \frac{13}{15}$$

$$P(E|A') = \frac{55}{500} = \frac{11}{100}, P(E|B') = \frac{55}{400} = \frac{11}{80}, P(E|\Gamma') = \frac{40}{300} = \frac{2}{15}$$

$$P(E'|A') = \frac{435}{500} = \frac{89}{100}, P(E'|B') = \frac{345}{400} = \frac{69}{80}, P(E'|\Gamma') = \frac{260}{300} = \frac{13}{15}$$

$$P(\Lambda|A) = \frac{50}{100} = \frac{1}{2}, P(\Lambda|B) = \frac{120}{200} = \frac{3}{5}, P(\Lambda|\Gamma) = \frac{200}{300} = \frac{2}{3}$$

$$P(\Lambda'|A) = \frac{50}{100} = \frac{1}{2}, P(\Lambda'|B) = \frac{80}{200} = \frac{2}{5}, P(\Lambda'|\Gamma) = \frac{100}{300} = \frac{1}{3}$$

$$P(\Lambda|A') = \frac{180}{500} = \frac{9}{25}, P(\Lambda|B') = \frac{150}{400} = \frac{3}{8}, P(\Lambda|\Gamma') = \frac{130}{300} = \frac{13}{30}$$

$$P(\Lambda'|A') = \frac{320}{500} = \frac{16}{25}, P(\Lambda'|B') = \frac{250}{400} = \frac{5}{8}, P(\Lambda'|\Gamma') = \frac{170}{300} = \frac{17}{30}$$

Ο τύπος του Bayes για τον υπολογισμό της πιθανότητας ενός αντικειμένου με συγκεκριμένα χαρακτηριστικά x να ανήκει στη κλάση i είναι ο εξής:

$$P(i|x) = \frac{P(i) * \prod_{k=1}^p P(x^k|i)}{\sum_{j=1}^c P(j) * \prod_{k=1}^p P(x^k|j)}$$

Θα χρησιμοποιήσουμε τον παραπάνω τύπο, καθώς και τους υπολογισμούς που κάναμε, προκειμένου να προσδιορίσουμε τη λειτουργία του ταξινομητή Naive-Bayes. Π.χ. για έναν λαμπτήρα που είναι ελαττωματικός και είναι λαμπτήρας αλογόνου έχουμε:

$$P(A|E, \Lambda) = \frac{P(A)*P(E|A)*P(\Lambda|A)}{P(A)*P(E|A)*P(\Lambda|A) + P(A')*P(E|A')*P(\Lambda|A')} = \frac{\frac{1}{6} * \frac{3}{20} * \frac{1}{2}}{\frac{1}{6} * \frac{3}{20} * \frac{1}{2} + \frac{5}{6} * \frac{11}{100} * \frac{9}{25}} = \frac{\frac{1}{80}}{\frac{1}{80} + \frac{33}{1000}} = \frac{25}{91}$$

$$P(A|E, \Lambda) \approx 0,275$$

$$P(B|E, \Lambda) = \frac{P(B)*P(E|B)*P(\Lambda|B)}{P(B)*P(E|B)*P(\Lambda|B) + P(B')*P(E|B')*P(\Lambda|B')} = \frac{\frac{1}{3} * \frac{1}{8} * \frac{3}{5}}{\frac{1}{3} * \frac{1}{8} * \frac{3}{5} + \frac{2}{3} * \frac{11}{80} * \frac{3}{8}} = \frac{\frac{1}{40}}{\frac{1}{40} + \frac{11}{32}} = \frac{4}{9}$$

$$P(B|E, \Lambda) \approx 0,44$$

$$P(\Gamma|E, \Lambda) = \frac{P(\Gamma)*P(E|\Gamma)*P(\Lambda|\Gamma)}{P(\Gamma)*P(E|\Gamma)*P(\Lambda|\Gamma) + P(\Gamma')*P(E|\Gamma')*P(\Lambda|\Gamma')} = \frac{\frac{1}{2} * \frac{2}{15} * \frac{2}{3}}{\frac{1}{2} * \frac{2}{15} * \frac{2}{3} + \frac{1}{2} * \frac{2}{15} * \frac{13}{30}} = \frac{\frac{2}{45}}{\frac{2}{45} + \frac{13}{450}} = \frac{20}{33}$$

$$P(\Gamma|E, \Lambda) \approx 0,606$$

Συνεπώς, μεγαλύτερη είναι η πιθανότητα $P(\Gamma|E, \Lambda)$ και ο λαμπτήρας προέρχεται από τον κατασκευαστή Γ.