Τεχνητή Νοημοσύνη

Εργασία 1 – Θεωρητικές Ασκήσεις
Κωνσταντίνος Πυθαρούλιος
Α.Μ.: 111520 1600142

Πρόβλημα 2:

Ο μικρότερος αριθμός κόμβων που μπορούν να δημιουργηθούν από τον αλγόριθμο είναι το 1.
Πρόκειται για την περίπτωση που η ρίζα του δένδρου αποτελεί τον κόμβο – στόχο.

Ο μεγαλύτερος αριθμός κόμβων που μπορούν να δημιουργηθούν από τον αλγόριθμο είναι b^d.
Αρκεί να σκεφτούμε ότι:

Οι κόμβοι σε βάθος d δημιουργούνται μία φορά.
Οι κόμβοι σε βάθος d-1 δημιουργούνται 2 φορές.
Οι κόμβοι σε βάθος d-2 δημιουργούνται 3 φορές.
Κοκ.
Μέχρι να φτάσουμε στο βάθος 1 που δημιουργείται d φορές.

Άρα έχουμε:

(d)*b + (d-1)*b^2 + (d-2)*b^3 + … + (2)*b^d-1 + (1)*b^d = O(b^d)

Πρόβλημα 3:

1. **Αναζήτηση πρώτα σε πλάτος (BFS).**
   (α). Κόμβος Στόχος: G1
   (b). Σειρά που βγαίνουν οι κόμβοι από το σύνορο(queue):
   S , A

Είναι σημαντικό να σημειωθεί ότι η μέθοδος που χρησιμοποιήθηκε είναι αυτή του αλγορίθμου της διαφάνειας νο. 27 από το πακέτο διαφανειών με τίτλο “Αναζήτηση σε γράφους”. Σύμφωνα με αυτόν, όταν ένας κόμβος εξέρχεται από την ουρά, ελέγχονται ένα ένα τα παιδιά του για να διαπιστωθεί αν κάποιο από αυτά αποτελεί κόμβο στόχο. Αν όντως, βρεθεί τέτοιος κόμβος, επιστρέφεται αμέσως χωρίς προτού να μπει στην ουρά.

2. Αναζήτηση πρώτα σε βάθος (DFS).

(α). Κόμβος Στόχος: G3

(b). Σειρά που βγαίνουν οι κόμβοι από το σύνορο(stack):
S, A, B, C, F, E, G3

Όσον αφορά τη σειρά που βγαίνουν οι κόμβοι, εδώ κύριο μου μέλλημα ήταν να δείξω την σειρά με την οποία ο αλγόριθμος επισκέπτεται τους κόμβους μέχρι να βρει τον κόμβο στόχο. Η επιλογή του επόμενου κόμβου κάθε φορά γίνεται αποκλειστικά βάση αλφαβητικής σειράς.

3. Αναζήτηση πρώτα σε βάθος με επαναληπτική εκβάθυνση.

i. Βάθος 0:

(α). Κόμβος Στόχος: Δεν βρέθηκε

(b). Σειρά που βγαίνουν οι κόμβοι από το σύνορο: S

ii. Βάθος 1:

(α). Κόμβος Στόχος: Δεν βρέθηκε

(b). Σειρά που βγαίνουν οι κόμβοι από το σύνορο:
S, A, B, D

iii. Βάθος 2:

(α). Κόμβος Στόχος: G1

(b). Σειρά που βγαίνουν οι κόμβοι από το σύνορο: S, A

Και εδώ, όπως και στο 1. παρατηρούμε ότι το G1 δεν παρουσιάζεται στο σύνορο αφού δεν προλαβαίνει να μπει σε αυτό. Αντίθετα, όταν το A βγει από το σύνορο επιτόπου, ελέγχονται όλοι οι κόμβοι-παιδιά του, προκειμένου να διαπιστωθεί αν κάποιος από αυτούς είναι κόμβος στόχος. Εδώ βρίσκουμε το G1, το οποίο επιστρέφεται χωρίς να εισέλθει προτού στο σύνορο.

4. Άπληστη αναζήτηση πρώτα στον καλύτερο

(α). Κόμβος Στόχος: G2

(b). Σειρά που βγαίνουν οι κόμβοι από το σύνορο: S, B, C

Εδώ το σύνορο είναι μια ουρά προτεραιότητας και βασιζόμαστε στις τιμές της ευρετικής συνάρτησης h.
Και εδώ, όταν εξάγεται ένας κόμβος από το σύνορο, αμέσως ελέγχονται τα παιδιά του για να διαπιστωθεί αν κάποιο αποτελεί κόμβο-στόχο. Έτσι, μετά την εξαγωγή του C, βρίσκουμε πως το C1 είναι κόμβος στόχος και επιστρέφεται χωρίς προτού να μπει στο σύνορο.

5. A*

(α). Κόμβος Στόχος: G2

(b). Σειρά που βγαίνουν οι κόμβοι από το σύνορο: S, D, C, G2

Εδώ, τώρα, για να βρούμε το G2, θα πρέπει να εξάγουμε προτού από το σύνορο ρα S, D ,C.
Να σημειωθεί πως ο κόμβος G2 έχει κόστος 13.

Πρόβλημα 4:

Ύστερα από μελέτη κάποιων περιπτώσεων καταλήγουμε στον εξής γενικό αλγόριθμο:

- Αν το μέγεθος της στοίβας είναι 1, δεν χρειάζεται κάποιο αναποδογύρισμα.
- Αν το μέγεθος της στοίβας είναι 2, ταξινομούμε τις δύο σφακιανές πίτες κάνοντας το πολύ ένα αναποδογύρισμα (αν είναι ανάποδα μεταξύ τους).
- Αλλιώς, αναποδογυρίζουμε τη n-οστή σφακιανή πίτα να φτάσει στην κορυφή και έπειτα κάνουμε άλλο ένα αναποδογύρισμα για να φτάσει στο τέλος της στίβας. Επαναλαμβάνουμε τον αλγόριθμο αυτόν n-1 φορές.

Οπότε, από τον πάνω αλγόριθμο προκύπτουν κάποιες περιπτώσεις μέγιστων αναποδογυρισμάτων μέχρι την ταξινόμηση της στοίβας. Έστω η συνάρτηση $T(n)$ που μας δείχνει τον αριθμό αναποδογυρισμάτων για n σφακιανές πίτες. Τότε:

- $T(1) = 0$
- $T(2) \leqslant 1$
- $T(n) \leqslant 2 + T(n-1)$, για $n \geqslant 3$

ή αλλιώς:

- **$T(1) = 0$**
- **$T(n) \leqslant 2n-3$, για $n \geqslant 2$**

I. Σύμφωνα με την παραπάνω ανάλυση έχουμε:

- $T(1) = 0$, οπότε **$f(1) = 0$**
- $T(2) \leqslant 1$, οπότε **$f(2) = 1$**
- $T(3) \leqslant 3$, οπότε **$f(3) = 3$**
- $T(4) \leqslant 4$, οπότε **$f(4) = 5$**

II. Όπως δείξαμε και νωρίτερα, αφού η συνάρτηση $f(n)$ μας δείχνει τον μικρότερο αριθμό αναποδογυρισμάτων για την χειρότερη περίπτωση εισόδου, εφόσον για **$n \geqslant 4$** έχουμε **$T(n) <= 2n-3$**, τότε **$f(n) = 2n-3$**.
Επομένως:

$n \geqslant 4 \Leftrightarrow$
$2n \geqslant 4+n \Leftrightarrow$ // προσθέσαμε n και στις δύο πλευρές
$2n-3 \geqslant n+1 \Leftrightarrow$ // αφαιρέσαμε 3 και από τις δύο πλευρές
$f(n) \geqslant n+1 \Leftrightarrow$
$f(n) \geqslant n$, που είναι το ζητούμενο για $n \geqslant 4$.

III. Θα ξεκινήσουμε από το συμπέρασμα ότι $f(n) \leqslant 2n$ (για $n \geqslant 1$) και θα καταλήξουμε σε κάτι που ισχύει πάντα.
Διακρίνουμε τις δύο ακόλουθες περιπτώσεις:

- **Για n = 1:**
  $f(n) \leqslant 2n \Rightarrow$
  $f(1) \leqslant 2*1 \Rightarrow$ // Το f(1) είναι γνωστό από το ερώτημα Ι.
  $0 \leqslant 2$, που ισχύει πάντα.
- **Για n ⩾ 2:**
  $f(n) \leqslant 2n \Rightarrow$
  $2n-3 \leqslant 2n \Rightarrow$ // Εφόσον $T(n) \leqslant 2n-3$, για $n \geqslant 2$ τότε $f(n) = 2n-3$, για $n \geqslant 2$.
  $-3 \leqslant 0$, που ισχύει πάντα.

Άρα, όντως, ισχύει ότι $f(n) \leqslant 2n$ για κάθε $n \geqslant 1$, όπου ο n είναι ακέραιος αριθμός.

IV. Τυπικός ορισμός:

- **Καταστάσεις:** μια περιγραφή κατάστασης καθορίζει τη θέση κάθε σφακιανής πίτας στη στοίβα.
- **Ενέργειες:** Μια σφακιανή πίτα αναποδογυρίζει με μια άλλη.
- **Αρχική κατάσταση:** Η στοίβα είναι άδεια.
- **Κατάσταση στόχου:** Είναι η κατάσταση που όλες οι σφακιανές πίτες είναι ταξινομημένες από πάνω προς τα κάτω στη στοίβα, από την μικρότερη στη μεγαλύτερη.
- **Κόστος μονοπατιού:** Το πλήθος των αναποδογυρισμάτων.
- **Μέγεθος χώρου καταστάσεων:** 2n-3, αφού τόσα είναι τα περισσότερα αναποδογυρίσματα που ενδέχεται να χρειαστούν.

Πρόβλημα 5

Πριν ξεκινήσουμε την οποιαδήποτε εξήγηση του προβλήματος είναι σημαντικό να δηλώσουμε την παραδοχή στην οποία έχουμε προβεί. Όπως μπορεί εύκολα να διαπιστώσει κανείς από το σχήμα που μας δίνεται, είναι αδύνατο τα δύο ανθρωπάκια να κινηθούν με οποιονδήποτε τρόπο ταυτόχρονα και να καταφέρουν να φτάσουν την ίδια ακριβώς στιγμή στα αντίστοιχα τετραγωνάκια-στόχους. Άρα, κάπως έτσι συμπεραίνουμε πως το πρόβλημα δεν είναι εφικτό να λυθεί. Επειδή, όμως, θέλουμε και λίγο να πειραματιστούμε, η μικρή παραδοχή στην οποία καταλήγουμε για να μπορεί τελικά το πρόβλημα να επιλυθεί είναι το να μπορεί ένα ανθρωπάκι πέρα από το να μετακινηθεί πάνω, κάτω, δεξιά ή αριστερά, να μπορεί να μένει και στάσιμο. Έτσι, η τεχνική που θα χρησιμοποιήσουμε είναι, όταν ένα ανθρωπάκι φτάσει μόλις πριν από το τετραγωνάκι στόχο, θα μείνει στάσιμο μέχρι να φτάσει και το άλλο μια θέση πριν τον στόχο του και τελικά στο τελευταίο βήμα να μεταβούν μαζί στο αντίστοιχο (και διαφορετικό) τετραγωνάκι-στόχο.

Ορισμός προβλήματος αναζήτησης

- **Καταστάσεις:** Ορίζονται τα τετραγωνάκια που βρίσκονται κάθε φορά τα ανθρωπάκια.
- **Ενέργεια:** Τα ανθρωπάκια κινούνται κάθε φορά αριστερά, δεξιά, πάνω, κάτω ή μένουν στάσιμα (παραδοχή).
- **Αρχική κατάσταση:** δίνεται κάθε φορά.
- **Κατάσταση στόχου:** Όλα τα ανθρωπάκια πάνω σε ένα διαφορετικό τετραγωνάκι-στόχο που συμβολίζεται με ευρώ.
- **Κόστος μονοπατιού:** Συνολικό κόστος διέλευσης τετραγώνων από την αρχική θέση που είναι το ανθρωπάκι μέχρι το τετραγωνάκι-στόχο (κόστος από λευκό τετραγωνάκι: 1, κόστος από καφέ τετραγωνάκι: 2).

Μέγεθος χώρου αναζήτησης

Με δεδομένο ότι έχουμε 2 ανθρωπάκια και αυτά έχουν πέντε δυνατές επιλογές για κάθε τους βήμα(αριστερά, δεξιά, πάνω, κάτω, στάσιμο), προκύπτουν 5^2 **πιθανοί συνδυασμοί σε κάθε βήμα** προς τα τετραγωνάκια στόχους.
Στην χειρότερη περίπτωση τώρα κάθε ανθρωπάκι μέχρι να φτάσει στο τετραγωνάκι-στόχο θα μπορούσε να διασχίσει όλα τα δυνατά τετραγωνάκια αν από αυτά αφαιρούσαμε τα τετραγωνάκια που αντιπροσωπεύουν τείχη αλλά και τα Κ τετραγωνάκια στα οποία βρίσκονται τα υπόλοια ανθρωπάκια όπως επίσης και το ίδιο, δηλαδή **n*n - αριθμός τετραγώνων τειχών - Κ αριθμός ανθρώπων.**

Οπότε στο συγκεκριμένο πρόβλημα των 64 τετραγώνων, με Κ=2 και 8 τετραγωνάκια-τείχη θα ισχύει: $(5^2)^{64-8-2} = (5^2)^{54} = 5^{108}$.

Γενικά: $(5^2)^{n*n-αριθμος τειχων-K}$

Μέγιστος παράγοντας διακλάδωσης

Ο μέγιστος παράγοντας διακλάδωσης είναι 5^{κ}, αν το ανθρωπάκι έχει τη δυνατότητα να πάει παντού (αριστερά, δεξιά, πάνω, κάτω, στάσιμο).

Βάθος βέλτιστης λύσης

Ουσιαστικά, πρόκειται για τον μικρότερο αριθμό βημάτων που μπορούν να γίνουν για να φτάσουν και τα δύο ανθρωπάκια στα τετράγωνα-στόχους. Το ελάχιστο για το πάνω ανθρωπάκι είναι 10 κουτάκια και για το κάτω 9, άρα η απάντηση είναι το 10.

Ευρετικές συναρτήσεις με Α*

- Μια καλή επιλογή θα ήταν να χρησιμοποιήσουμε **Απόσταση Manhattan**. Γενικά, σε περιπτώσεις που μπορούμε να κινηθούμε αριστερά, δεξιά, πάνω και κάτω, αυτή η ευρετική συνάρτηση μπορεί να είναι πολύ χρήσιμη. Υπολογίζει το άθροισμα του απόλυτου της διαφοράς των συντεταγμένων x,y της τωρινής και της τελικής κατάστασης.

- Μια άλλη τεχνική, η οποία όμως μπορεί να αποβεί πολύ χρονοβόρα, είναι να χρησιμοποιήσουμε **Ευκλείδια Απόσταση** για να υπολογίσουμε την ακριβή απόσταση δύο τετραγώνων.