ΜΟΝΤΕΛΟΠΟΙΗΣΗ ΠΡΟΒΛΗΜΑΤΩΝ ΑΝΑΖΗΤΗΣΗΣ

Ένα **πρόβλημα τεχνητής νοημοσύνης** απαιτεί για την **μοντελοποίηση** του ως πρόβλημα **αναζήτησης** τα εξής:

1. Κατάσταση (μία μαθηματική αναπαράσταση ενός στιγμιότυπου του προβλήματος).

Συνήθως είναι ένας **πίνακας** (π.χ. μονοδιάστατος, διδιάστατος κ.λπ.) με δυαδική, ακέραια ή κωδικοποίηση πραγματικών αριθμών με τις απαραίτητες πληροφορίες ενός στιγμιότυπου.

Π.χ. στο σκάκι είναι ένας 8x8 πίνακας που απεικονίζεται η θέση των πεσσών, στο λαβύρινθο η θέση του ρομπότ κ.λπ.

2. Τελεστές Μετάβασης (Μηχανισμός Αλλαγής Καταστάσεων) – οι κινήσεις που επιτρέπονται στο πρόβλημα ως ενέργειες σε μια κατάσταση

Συντακτικό Τελεστών Μετάβασης:

Όνομα_Τελεστή(πιθανά ορίσματα): Περιγραφή Ενέργειας
Προϋποθέσεις: Καταγραφή Συνθηκών που πρέπει να ισχύουν
Αποτέλεσμα: Αλλαγές που επέρχονται στην κατάσταση του προβλήματος

3. Συνάρτηση Πραγματικού Κόστους g(v) (το κόστος από την αφετηρία έως τον κόμβο v).

Είναι πάντα το άθροισμα των βαρών από την αφετηρία έως και τον τρέχοντα κόμβο.

Θέτουμε (συνήθως) τα βάρη των ακμών ίσα με 1 (ισοδύναμα το κόστος εφαρμογής των τελεστών μετάβασης), οπότε η συνάρτηση πραγματικού κόστους g(v) είναι το άθροισμα των βαρών των ακμών από την αφετηρία έως τον κόμβο v

4. Ευρετική Συνάρτηση h(v) (η εκτίμηση για την απόσταση του v από έναν κόμβο-στόχο).

Μία συνάρτηση που δίνει έναν αριθμό σε μία κατάσταση.
Όσο πιο μικρός ο αριθμός τόσο καλύτερος ο κόμβος (υπό την έννοια ότι εκτιμάται ότι απέχει λιγότερο από την αφετηρία)
Ο κόμβος στόχος έχει πάντα τιμή 0.

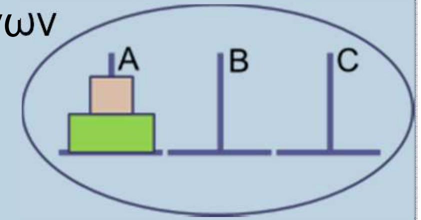
Μία ευρετική συνάρτηση καλείται **παραδεκτή** αν δεν υπερεκτιμάει το πραγματικό κόστος (δηλ. $h(v) \leq h^*(v)$ για κάθε κόμβο του γραφήματος).

ΧΩΡΟΣ ΚΑΤΑΣΤΑΣΕΩΝ (ή ΓΡΑΦΟΣ ΚΑΤΑΣΤΑΣΕΩΝ)

<u>Χώρος Καταστάσεων (ή Γράφος Καταστάσεων)</u>: Είναι ένας γράφος που για κορυφές έχει όλες τις έγκυρες καταστάσεις ενός προβλήματος αναζήτησης και για ακμές έχει τους τελεστές μετάβασης

<u>Παράδειγμα Κατασκευής Χώρου Καταστάσεων:</u> Σε μία εκδοχή του προβλήματος των Πύργων του Hanoi υπάρχουν 3 πύργοι (A, B, C) και 2 δίσκοι (μικρός και μεγάλος). Στόχος είναι να μεταφερθούν και οι 2 δίσκοι από τον πύργο A στον πύργο C σύμφωνα με τα παρακάτω:

- Μπορεί να μεταφερθεί μόνο ένα δίσκος τη φορά.
- Δε μπορεί να τοποθετηθεί ο μεγάλος δίσκος πάνω από τον μικρό δίσκο.



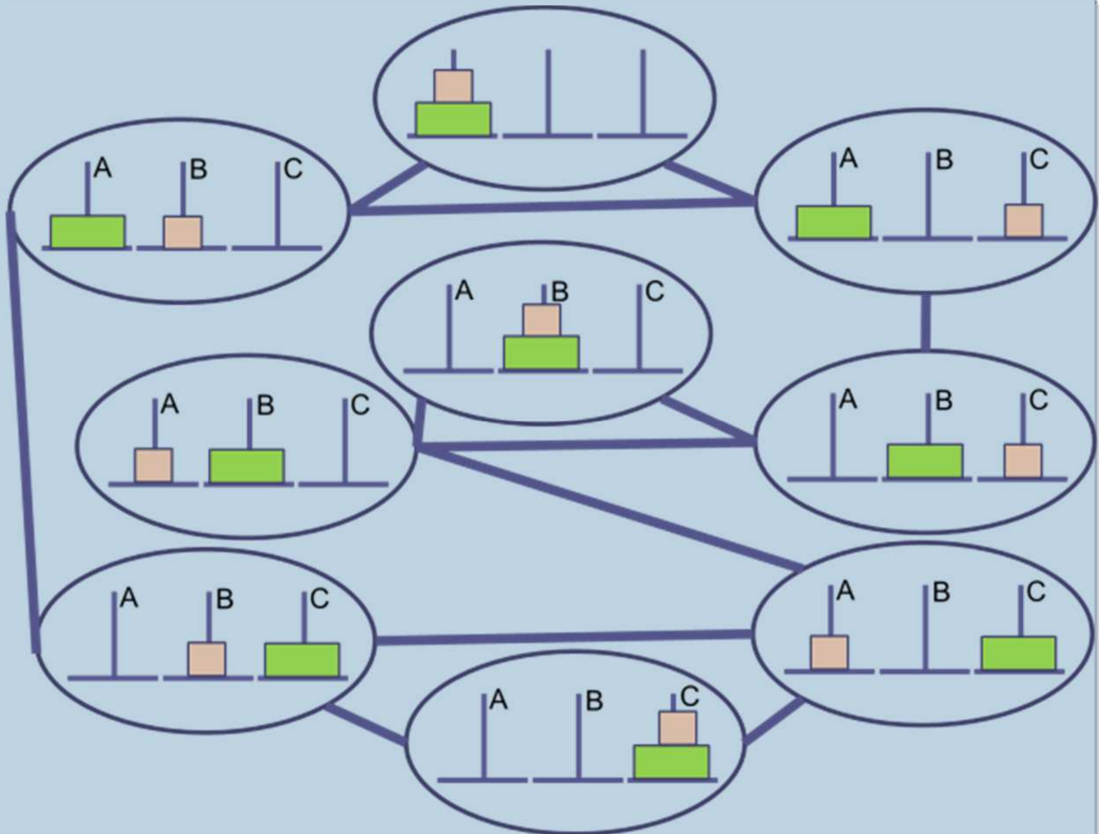
<u>Λύση:</u>

<u>Κατάσταση:</u> Αναπαριστώ μια κατάσταση με μια διάδα (X,Y) όπου X∈{A,B,C} δείχνει σε ποιο πύργο βρίσκεται ο μεγάλος δίσκος και Y∈{A,B,C} δείχνει σε ποιο πύργο βρίσκεται ο μικρός δίσκος

<u>Τελεστές</u>.

T_1(x): Μετακίνηση του μικρού δίσκου στον πύργο x
<u>Προϋποθέσεις:</u> Ο μικρός δίσκος βρίσκεται σε πύργο διαφορετικό του x
<u>Αποτέλεσμα:</u> Ο μικρός δίσκος μεταφέρεται στον πύργο x

T_2(x): Μετακίνηση του μεγάλου δίσκου στον πύργο x
<u>Προϋποθέσεις:</u> Ο μεγάλος δίσκος βρίσκεται σε πύργο διαφορετικό του x, ο μικρός δίσκος δεν βρίσκεται στον πύργο x και δεν είναι επάνω στον μεγάλο δίσκο
<u>Αποτέλεσμα:</u> Ο μεγάλος δίσκος μεταφέρεται στον πύργο x



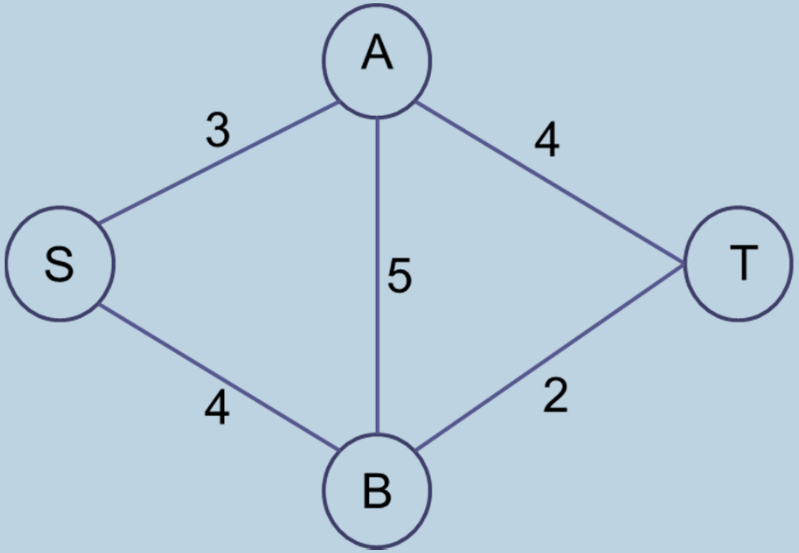
Ο γράφος καταστάσεων ορίζεται άμεσα από τον ορισμό της κατάστασης (κόμβοι) και τους τελεστές (ακμές). Ισχύει ότι τα βάρη των ακμών είναι το κόστος των τελεστών (εδώ π.χ. ίσα με 1) και ότι με μη κατευθυνόμενες ακμές ορίζουμε την εφαρμογή τελεστών αμφίδρομα.

ΧΩΡΟΣ ΑΝΑΖΗΤΗΣΗΣ

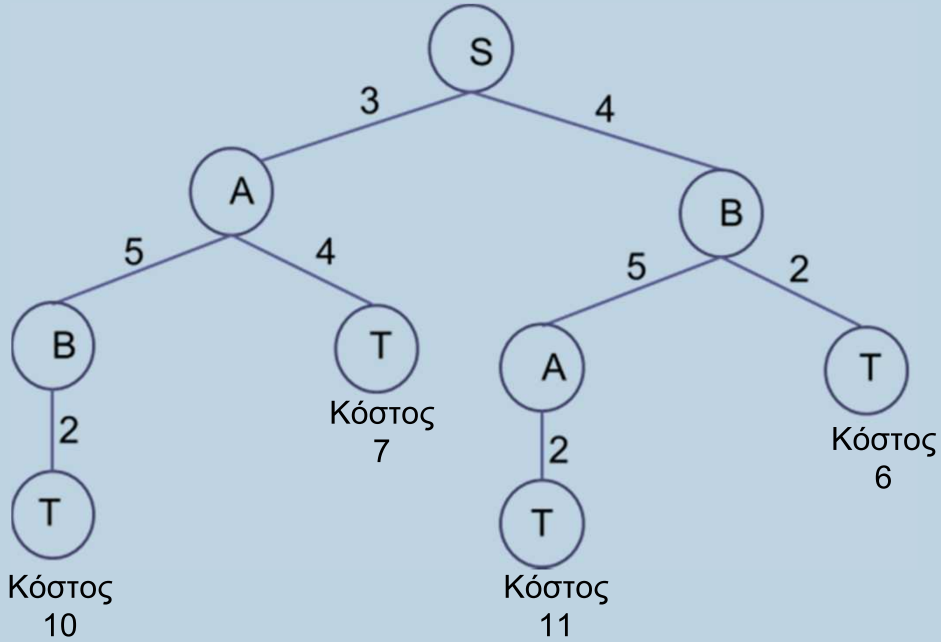
<u>Χώρος Αναζήτησης</u>: Δένδρο που περιλαμβάνει όλα τα μονοπάτια από την αφετηρία έως το κόμβο-στόχο.

Κατασκευή χώρου αναζήτησης: «Θέσε ως παιδιά του κόμβου, τους γείτονες στο γράφο, που δεν είναι πρόγονοι. Σταμάτα όταν είσαι σε αδιέξοδο ή στον τερματισμό»

<u>Παράδειγμα Κατασκευής Χώρου Αναζήτησης:</u> Στον ακόλουθο γράφο καταστάσεων με αφετηρία τον κόμβο S και τερματισμό το T, δώστε το χώρο αναζήτησης του προβλήματος.



<u>Χώρος Καταστάσεων</u>



<u>Χώρος Αναζήτησης</u>

<u>Στον χώρο αναζήτησης (με τα βάρη):</u>

<u>Κόστος μονοπατιού:</u> Άθροισμα βαρών ακμών από αφετηρία έως κόμβο (π.χ. κόστος S-A-B-T είναι 10)

<u>Βέλτιστο μονοπάτι:</u> Το συντομότερο (μικρότερο κόστος) από όλα τα μονοπάτια (εδώ είναι το S-B-T με κόστος 6)

ΑΝΑΖΗΤΗΣΗ ΚΑΤΑ ΒΑΘΟΣ

Ο **αλγόριθμος Αναζήτησης Κατά Βάθος** επιστρέφει ένα μονοπάτι από την αφετηρία προς τον τερματισμό σε έναν γράφο αναζήτησης (Αλγόριθμος *Τυφλής Αναζήτησης* διότι δεν κοιτάει βάρη ακμών – ευρετικές εκτιμήσεις).

ΑΛΓΟΡΙΘΜΟΣ ΚΑΤΑ ΒΑΘΟΣ

Αρχικά:

Βάζουμε την αφετηρία στο δένδρο.

Επαναληπτικά:

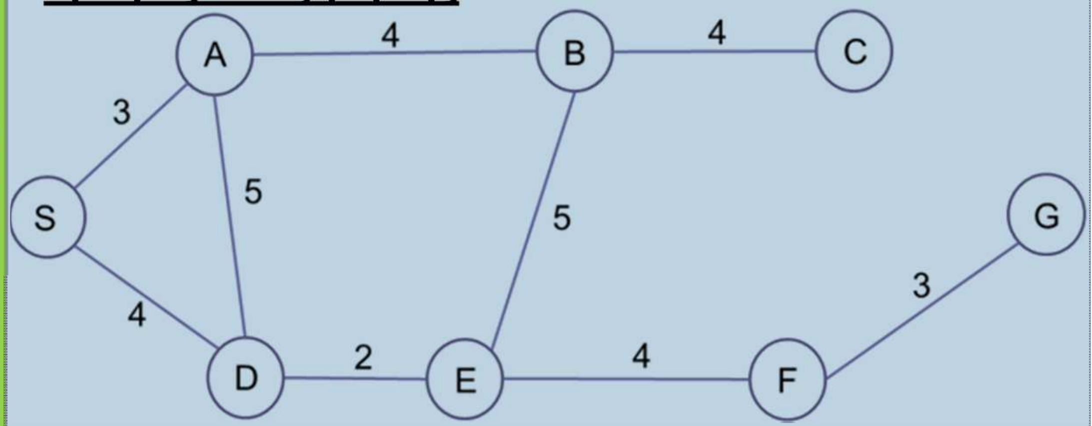
- **Πατάμε** «***αριστερόστροφα***» στον επόμενο κόμβο.
- **Ανοίγουμε** τους γείτονές στο γράφο (που δεν είναι πρόγονοί στο δένδρο) και τους θέτουμε ως παιδιά του
- **Διαγράφουμε** ανοικτές εμφανίσεις πατημένων κόμβων

Εως ότου:

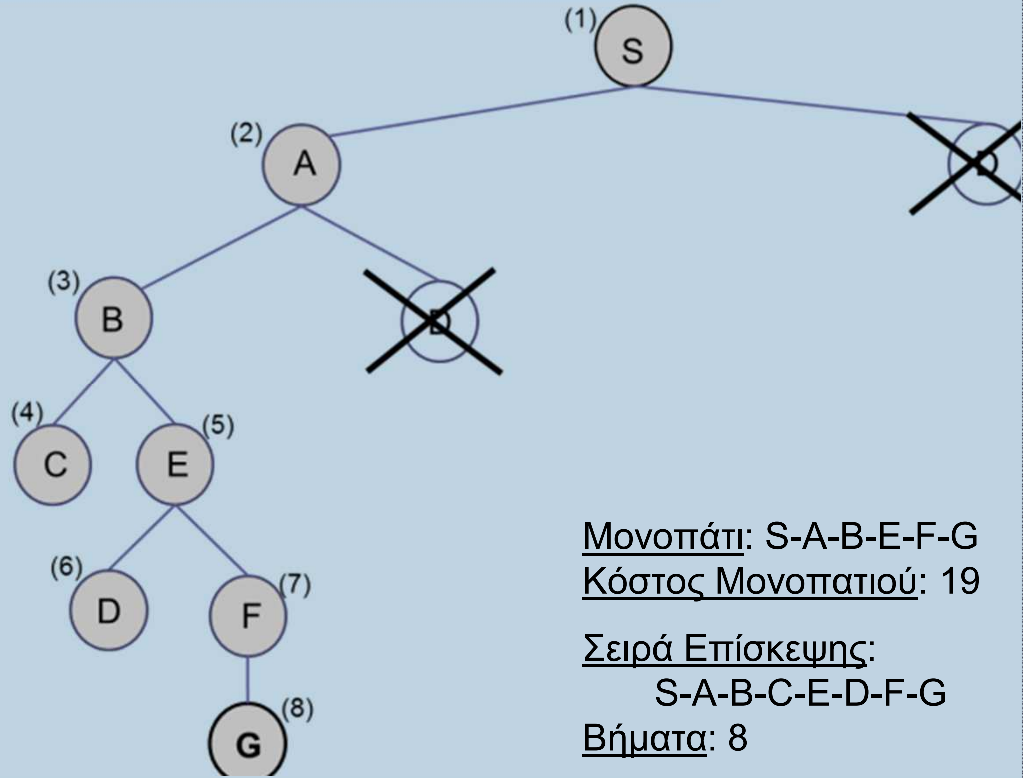
Πατήσουμε στον κόμβο-στόχο

Ορολογία:
Πάτημα=Επίσκεψη
Ανοιγμα=Επέκταση

Γράφος Αναζήτησης



Δένδρο Αναζήτησης της Κατά Βάθος



Μονοπάτι: S-A-B-E-F-G
Κόστος Μονοπατιού: 19

Σειρά Επίσκεψης:
S-A-B-C-E-D-F-G
Βήματα: 8

Πληρότητα: ΌΧΙ
Βελτιστότητα: ΌΧΙ
Πολ/τα χώρου:
Γραμμική: O(bd)
Πολ/τα χρόνου:
Εκθετική: $O(b^d)$

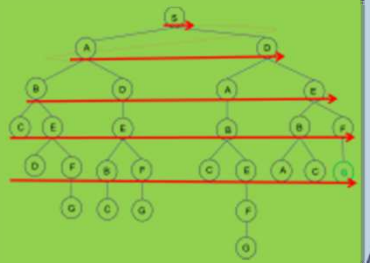
Κριτήρια Επιλογής:
(α) Περιορισμένος χώρος μνήμης. (β) Σε ένα επίπεδο έχουμε μόνο λύσεις

Υλοποίηση με στοίβα

Βήμα	ΑΝΟΙΚΤΕΣ	ΚΛΕΙΣΤΕΣ
0	(S)	{}
1	(A,D)	{S}
2	(B,D_{SA},D_S)	{S,A}
3	(C,E,D_{SA},D_S)	{S,A,B}
4	(E,D_{SA},D_S)	{S,A,B,C}
5	(D_{SABE},F,D_{SA},D_S)	{S,A,B,C,E}
6	(F)	{S,A,B,C,E,D}
7	(G)	{S,A,B,C,E,D,F}
8	()	{S,A,B,C,E,D,F,G}

Ο **αλγόριθμος Αναζήτησης Κατά Πλάτος** επιστρέφει ένα μονοπάτι από την αφετηρία προς τον τερματισμό σε έναν γράφο αναζήτησης (Αλγόριθμος *Τυφλής Αναζήτησης* διότι δεν κοιτάει βάρη ακμών – ευρετικές εκτιμήσεις).

ΑΛΓΟΡΙΘΜΟΣ ΚΑΤΑ ΠΛΑΤΟΣ



Αρχικά:

Βάζουμε την αφετηρία στο δένδρο.

Επαναληπτικά:

- **Πατάμε** «***κατά επίπεδα***» στον επόμενο κόμβο.
- **Ανοίγουμε** τους γείτονές στο γράφο (που δεν είναι πρόγονοί στο δένδρο) και τους θέτουμε ως παιδιά του
- **Διαγράφουμε** ανοικτές εμφανίσεις πατημένων κόμβων

Εως ότου:

Πατήσουμε στον κόμβο-στόχο

Ορολογία:
Πάτημα=Επίσκεψη
Ανοιγμα=Επέκταση

Πληρότητα: ΝΑΙ

Βελτιστότητα: *Μόνο* για ισοβαρείς ακμές

Πολ/τα χώρου:
Εκθετική: $O(b^d)$

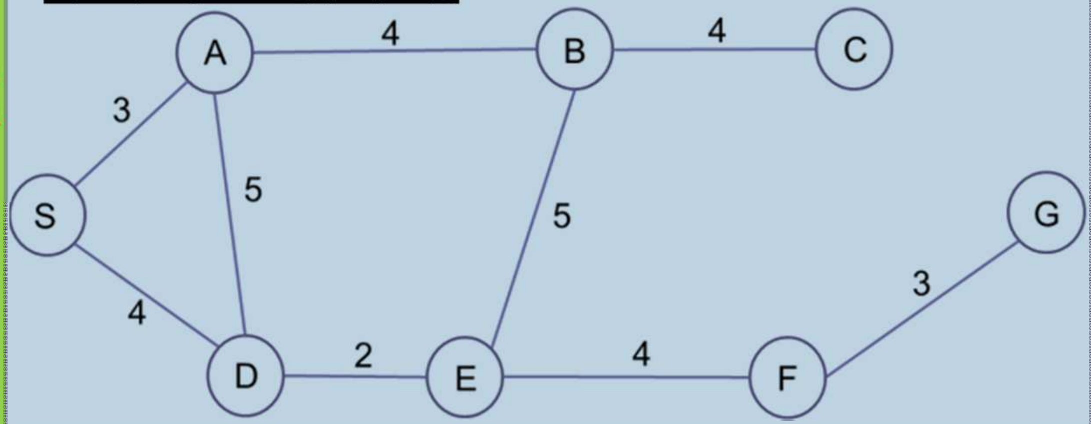
Πολ/τα χρόνου:
Εκθετική: $O(b^d)$

Κριτήρια Επιλογής:
(α) Βρίσκει τη λύση που είναι πιο κοντά στη ρίζα. (β) Βέλτιστη λύση (αν οι ακμές εχουν ίσο βάρος)

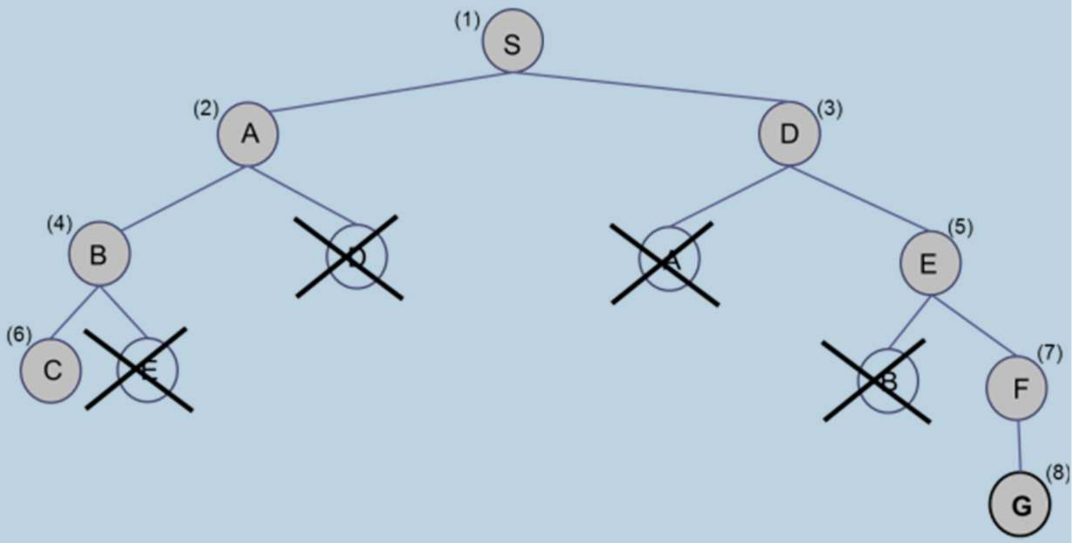
Υλοποίηση με ουρά

Βήμα	ΑΝΟΙΚΤΕΣ	ΚΛΕΙΣΤΕΣ
0	(S)	{}
1	(A,D)	{S}
2	(D_S,B,D_{SA})	{S,A}
3	(B,E)	{S,A,D}
4	(E_{SD},C,E_{SAB})	{S,A,D,B}
5	(C,F)	{S,A,D,B,E}
6	(F)	{S,A,D,B,E,C}
7	(G)	{S,A,D,B,E,C,F}
8	()	{S,A,D,B,E,C,F,G}

Γράφος Αναζήτησης



Δένδρο Αναζήτησης της Κατά Πλάτος



Μονοπάτι: S-D-E-F-G

Κόστος Μονοπατιού: 13

Σειρά Επίσκεψης: S-A-D-B-E-C-F-G

Βήματα: 8

Ο **αλγόριθμος ευρετικής αναζήτησης** επιστρέφει ένα μονοπάτι από την αφετηρία προς τον τερματισμό σε έναν γράφο αναζήτησης. Έχει τρεις παραλλαγές ανάλογα με τη **συνάρτηση αξιολόγησης f** που χρησιμοποιούμε.

ΕΥΡΕΤΙΚΗ ΑΝΑΖΗΤΗΣΗ

Αρχικά:

- Βάζουμε την αφετηρία στο δένδρο με τιμή **f(s)**.

Επαναληπτικά:

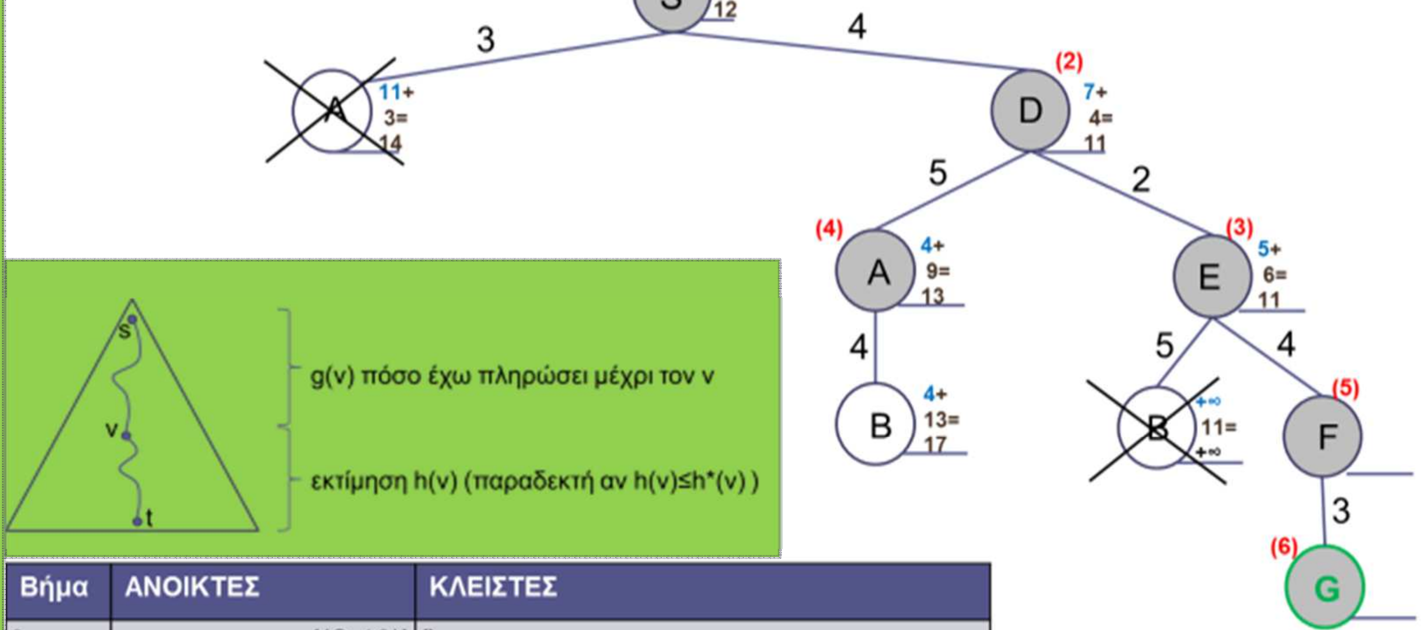
- **(Π)ατάμε** στον ανοιχτό κόμβο με την μικρότερη τιμή ***f(x)***.
- **(Α)νοίγουμε** τους γείτονες του στον γράφο (που δεν είναι πρόγονοί του στο δένδρο) και τους θέτουμε ως παιδιά του
- **(Α)ξιολογούμε** τα παιδιά με τη συνάρτηση **f(v)**:
- Για κάθε παιδί κρατάμε την καλύτερη (μικρότερη) εμφάνιση στο δένδρο. Αν βγήκε με μικρότερη τιμή το κρατάμε (και **(Δ)ιαγράφουμε** άλλες εμφανίσεις του στο δένδρο). Αλλιώς το διαγράφουμε.

Εως ότου:

- Πατήσουμε στον κόμβο-στόχο

UCS (**f=g** πραγματικό), **Greedy** (**f=h** ευρετικό), **A***(**f=g+h** Πραγμ+Ευρετ)

Παράδειγμα Δένδρου A*



Βήμα	ΑΝΟΙΚΤΕΣ	ΚΛΕΙΣΤΕΣ
0	{(S,12)}	{}
1	{(A,14),(D,11)}	{(S,12)}
2	{(A,13),(E,11)}	{(S,12),(D,11)}
3	{(A,13),(B,+∞)(F,13)}	{(S,12),(D,11),(E,11)}
4	{(B,17),(F,13)}	{(S,12),(D,11),(E,11),(A,13)}
5	{(B,17),(G,13)}	{(S,12),(D,11),(E,11),(A,13),(F,13)}
6	{(B,17)}	{(S,12),(D,11),(E,11),(A,13),(F,13),(G,13)}

Μονοπάτι: S-D-E-F-G
Κόστος Μονοπατιού: 13

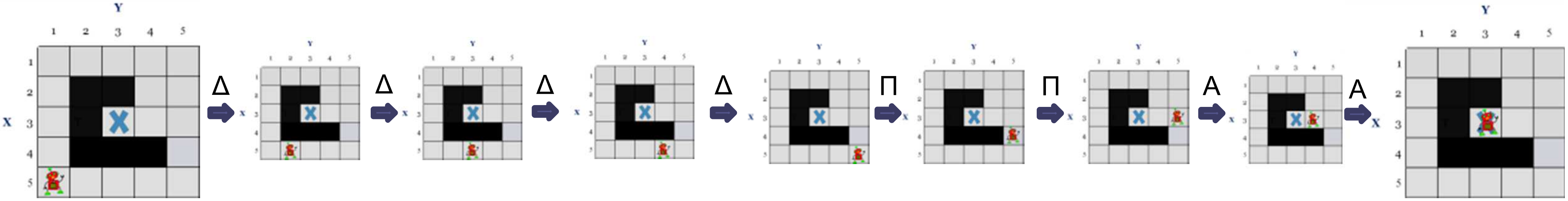
Σειρά Επίσκεψης:
S-D-A-E-F-G
Βήματα: 6

Όνομα Αλγορίθμου	**Πλήρης;**	**Βέλτιστος;**	**Χρόνος;**	**Χώρος;**	**Σχόλια;**
UCS (Αναζήτηση Ομοιόμορφου Κόστους)	ΝΑΙ	ΝΑΙ	Εκθετικός	Εκθετικός	Πολύ χρονοβόρος
Greedy(Άπληστος, Πρώτα στο Καλύτερο)	ΝΑΙ	ΌΧΙ	Εκθετικός	Εκθετικός	Άπληστος ;-)
A*	ΝΑΙ	*	Εκθετικός	Εκθετικός	Ιδανικός αλγόριθμος ΤΝ

*: Μόνο αν η ευρετική που χρησιμοποιείται είναι παραδεκτή

Το Πρόβλημα του Λαβυρίνθου

Στο **πρόβλημα του λαβυρίνθου** ένα ρομπότ βρίσκεται σε ένα τετράγωνο και πρέπει να μετακινηθεί σε ένα τετράγωνο-στόχο. Επιτρεπτές κινήσεις: Πάνω, Κάτω, Αριστερά, Δεξιά.



Κατάσταση: Αναπαριστούμε μία **κατάσταση** του προβλήματος με ένα **διατεταγμένο ζεύγος (X,Y)** όπου $X, Y \in \{1,2,3,4,5\}$ είναι οι συντεταγμένες που βρίσκεται το ρομπότ.

Τελεστές Δράσης: Ορίζουμε τους ακόλουθους 4 **τελεστές** οι οποίοι περιγράφουν τις κινήσεις που μπορεί να κάνει το ρομπότ:

ΠΑΝΩ: Μετακίνηση του ρομπότ μία θέση πάνω
Προϋποθέσεις: $X \neq 1$ και το τετράγωνο $(X-1, Y)$ δεν είναι εμπόδιο.
Αποτέλεσμα: Το ρομπότ μετακινείται στο τετράγωνο $(X-1, Y)$

ΚΑΤΩ: Μετακίνηση του ρομπότ μία θέση κάτω
Προϋποθέσεις: $X \neq 5$ και το τετράγωνο $(X+1, Y)$ δεν είναι εμπόδιο.
Αποτέλεσμα: Το ρομπότ μετακινείται στο τετράγωνο $(X+1, Y)$

ΑΡΙΣΤΕΡΑ: Μετακίνηση του ρομπότ μία θέση αριστερά
Προϋποθέσεις: $Y \neq 1$ και το τετράγωνο $(X, Y-1)$ δεν είναι εμπόδιο.
Αποτέλεσμα: Το ρομπότ μετακινείται στο τετράγωνο $(X, Y-1)$

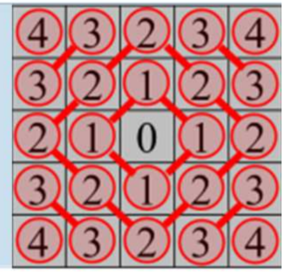
ΔΕΞΙΑ: Μετακίνηση του ρομπότ μία θέση δεξιά
Προϋποθέσεις: $Y \neq 5$ και τετράγωνο $(X, Y+1)$ δεν είναι εμπόδιο.
Αποτέλεσμα: Το ρομπότ μετακινείται στο τετράγωνο $(X, Y+1)$

Συνάρτηση Πραγματικού Κόστους: Ορίζουμε ότι το **κόστος κάθε ακμής** είναι ίσο με 1 (ισοδύναμα το κόστος εφαρμογής των τελεστών μετάβασης είναι ίσο με 1).
g(n): Άθροισμα βαρών των ακμών από την αφετηρία έως τον κόμβο n.

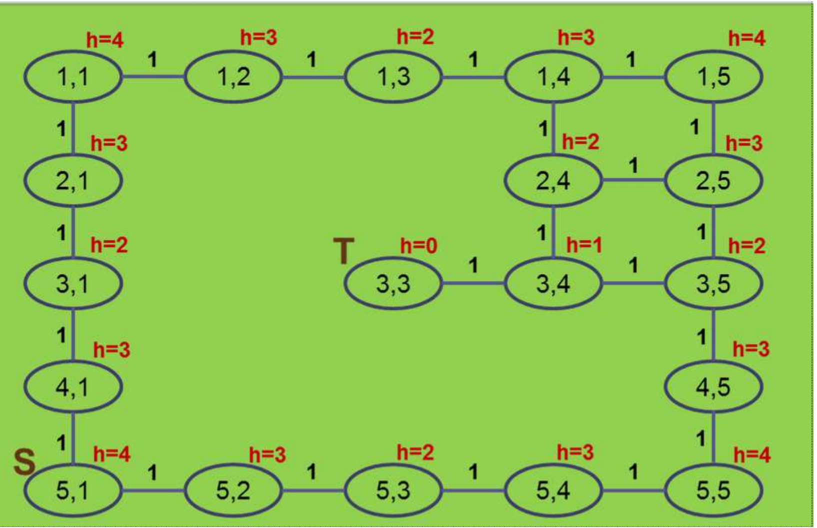
Ευρετική Συνάρτηση: Ορίζουμε ως ευρετική συνάρτηση την απόσταση Manhattan της κατάστασης (X,Y), από την κατάσταση-στόχο (X1,Y1):

manhattan ((X,Y),(X1,Y1)) = |X-X1|+|Y-Y1|

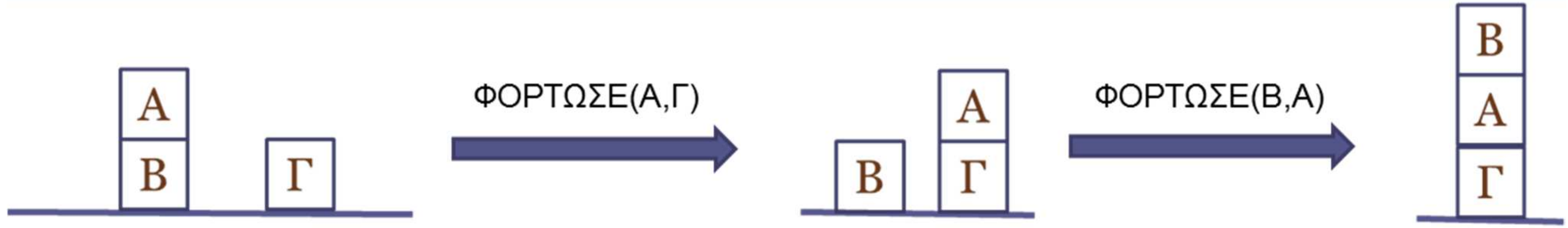
Παρατήρηση: Η ευρετική είναι **παραδεκτή**



Γράφος Καταστάσεων:



Στον **κόσμο των κύβων**, επιτρέπεται να μετακινήσουμε έναν κύβο εφόσον δεν έχει άλλο κύβο πάνω του είτε πάνω στο τραπέζι είτε πάνω σε μία στοίβα κύβων. Η σειρά των στοιβών (από κύβους) στο τραπέζι δεν έχει σημασία, ενώ η σειρά των κύβων στις στοίβες έχει σημασία. Ζητείται να κάνουμε τις μετακινήσεις των κύβων ώστε να πάμε από μία αρχική σε μία τελική κατάσταση.



Κατάσταση: Αναπαριστούμε μία **κατάσταση** του προβλήματος με ένα **σύνολο στοιβών,** όπου κάθε στοίβα αναπαρίσταται με μία **διατεταγμένη n-άδα** με τα ονόματα των στοιβών όπως βρίσκονται στη στοίβα από πάνω προς τα κάτω.

Για παράδειγμα η αρχική και η τελική κατάσταση αναπαριστώνται ως εξής: {(Α,Β),(Γ)} και {(Β,Α,Γ)} αντίστοιχα

Τελεστές Δράσης: Ορίζουμε τους τελεστές ΦΟΡΤΩΣΕ(Χ,Υ) και ΞΕΦΟΡΤΩΣΕ(Χ) ως εξής:

ΦΟΡΤΩΣΕ(Χ,Υ): Μετακίνηση του κύβου Χ πάνω στον κύβο Υ

Προϋποθέσεις:

(1) Ο κύβος Χ δεν έχει άλλο κύβο πάνω του

(2) Ο κύβος Υ δεν έχει άλλο κύβο πάνω του

Αποτέλεσμα:

Ο κύβος Χ είναι πάνω στον κύβο Υ

ΞΕΦΟΡΤΩΣΕ(Χ): Μετακίνηση του κύβου Χ στο τραπέζι

Προϋποθέσεις:

(1) Ο κύβος Χ δεν είναι στο τραπέζι

(2) Ο κύβος Χ δεν έχει άλλο κύβο πάνω του

Αποτέλεσμα:

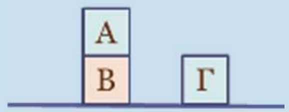
Ο κύβος Χ είναι στο τραπέζι

Συνάρτηση Πραγματικού Κόστους: Ορίζουμε ότι το **κόστος κάθε ακμής** είναι ίσο με 1 (ισοδύναμα το κόστος εφαρμογής των τελεστών μετάβασης είναι ίσο με 1).

g(n): Άθροισμα βαρών των ακμών από την αφετηρία έως τον κόμβο n.

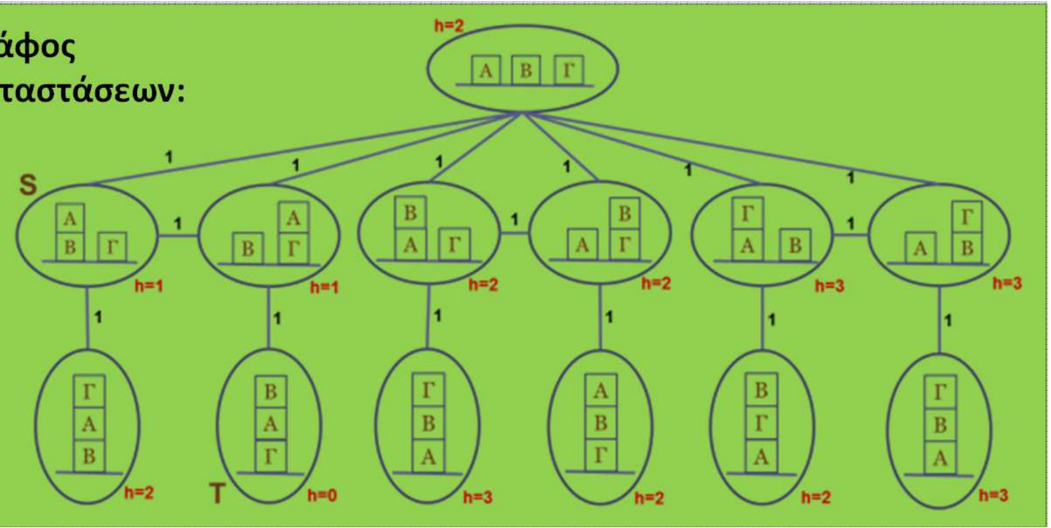
Ευρετική Συνάρτηση: Ορίζουμε ως ευρετική συνάρτηση, το πλήθος των κύβων που είναι σε **λάθος ύψος** σε σχέση με την κατάσταση-στόχο.

Παρατήρηση: Η ευρετική είναι **παραδεκτή**

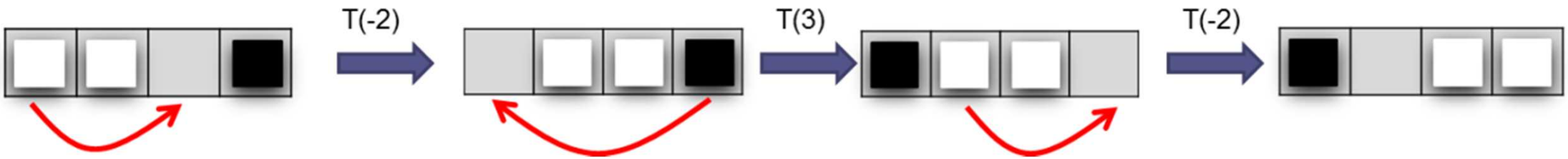


h({(Α,Β),(Γ)})=1

Γράφος Καταστάσεων:



Στο **Ευθύγραμμο Παζλ**, δίδεται ένα πλαίσιο 4 κενών θέσεων στο οποίο τοποθετούνται 3 πλακίδια εκ των οποίων τα δύο είναι άσπρα και το ένα είναι μαύρο. Οι κινήσεις που επιτρέπονται είναι μετακίνηση του πλακιδίου στην κενή θέση (δεξιά ή αριστερά) είτε απ΄ευθείας εφόσον είναι δίπλα του, είτε υπερπηδώντας άλλα πλακίδια.



<u>Κατάσταση:</u> Αναπαριστούμε μία **κατάσταση** του προβλήματος με έναν πίνακα 4 θέσεων που περιέχει τα γράμματα Λ (δύο φορές), Μ (μία φορά), Κ(συμβολίζει το κενό).

Για παράδειγμα η αρχική και η τελική κατάσταση αναπαριστώνται ως εξής: [Λ,Λ,Κ,Μ] και [Μ,Κ,Λ,Λ]

<u>Τελεστές Δράσης:</u> Ορίζουμε έναν τελεστή Τ(Χ) που συμβολίζει την μετακίνηση του κενού!

Θεωρώντας ότι το κενό είναι στη θέση $\Upsilon \in \{1,2,3,4\}$, έχουμε ότι:

Τ(Χ): Μετακίνηση του κενού Χ θέσεις {-3,-2,-1: Αριστερά, 1,2,3: Δεξιά}
<u>Προϋποθέσεις:</u> $1 \leq \Upsilon + X \leq 4$
<u>Αποτέλεσμα:</u> Το πλακίδιο στην θέση Υ+Χ μετακινείται στη θέση του κενού.

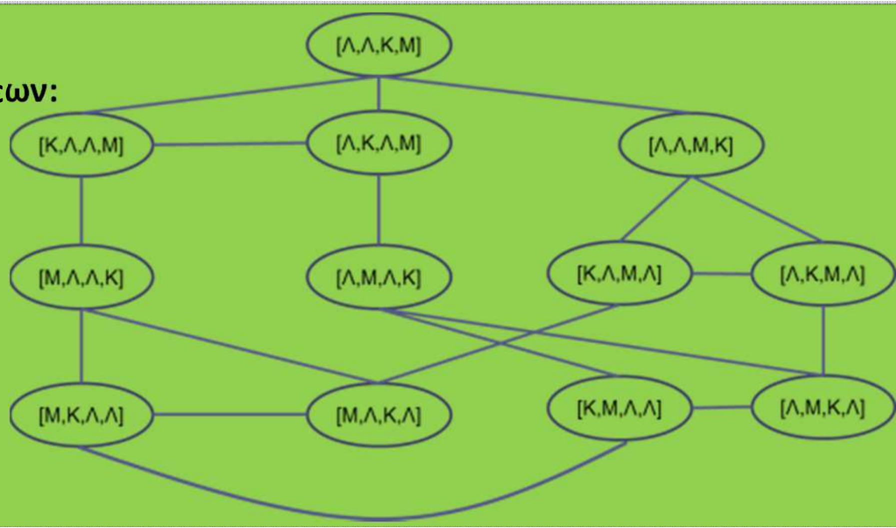
<u>Συνάρτηση Πραγματικού Κόστους:</u> Ορίζουμε ότι το **κόστος κάθε ακμής** είναι ίσο με 1 (ισοδύναμα το κόστος εφαρμογής των τελεστών μετάβασης είναι ίσο με 1).
g(n): Άθροισμα βαρών των ακμών από την αφετηρία έως τον κόμβο n.

<u>Ευρετική Συνάρτηση:</u> Ορίζουμε ως ευρετική συνάρτηση, το **πλήθος των πλακιδίων** που είναι σε **λάθος θέση** σε σχέση με την κατάσταση-στόχο.

Παρατήρηση: Η ευρετική είναι **παραδεκτή**

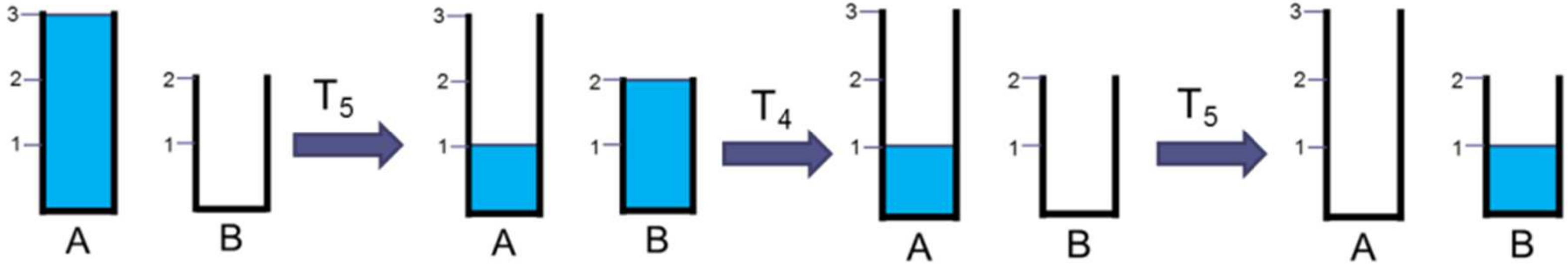
h([Λ,Λ,Κ,Μ])=3

Γράφος Καταστάσεων:



Το Πρόβλημα των Δοχείων

Στο **Πρόβλημα των Δοχείων** δίνονται δύο δοχεία Α και Β με χωρητικότητα 3 lt και 2 lt αντίστοιχα. Επιτρέπεται να γεμίσουμε (πλήρως) ένα δοχείο από τη βρύση, να αδειάσουμε ένα δοχείο ή να αδειάσουμε (όσο χωράει) από το ένα δοχείο στο άλλο.



<u>Κατάσταση:</u> Αναπαριστούμε μία **κατάσταση** του προβλήματος με ένα διατεταγμένο ζέυγος **(X,Y)** όπου X είναι τα λίτρα στο δοχείο Α και Y είναι τα λίτρα στο δοχείο Β.
Για παράδειγμα η αρχική και η τελική κατάσταση αναπαριστώνται ως εξής: **(3,0)** και **(0,1)** αντίστοιχα.

<u>Τελεστές Δράσης:</u> Ορίζουμε τους ακόλουθους 6 **τελεστές** που μοντελοποιούν τις επιτρεπτές ενέργειες:

T_1: Γέμισε το δοχείο Α
<u>Προϋποθέσεις:</u>
Το δοχείο Α δεν είναι γεμάτο (X≠3)
<u>Αποτέλεσμα:</u>
Το δοχείο Α είναι γεμάτο (X=3)

T_3: Άδειασε το δοχείο Α
<u>Προϋποθέσεις:</u>
Το δοχείο Α δεν είναι άδειο (X≠0)
<u>Αποτέλεσμα:</u>
Το δοχείο Α είναι άδειο (X=0)

T_5: Άδειασε το δοχείο Α στο δοχείο Β
<u>Προϋποθέσεις:</u>
(1) Το δοχείο Α δεν είναι άδειο (X≠0)
(2) Το δοχείο Β δεν είναι γεμάτο (Y ≠2)
<u>Αποτέλεσμα:</u>
Αδειάζουμε (όσο χωράει) από το Α στο Β
Αν |X+Y|≤2 τότε νέα κατάσταση: (0,X+Y)
Αν |X+Y|>2 τότε νέα κατάσταση: (X-(2-Y),2)

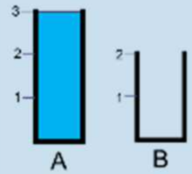
T_2: Γέμισε το δοχείο Β
<u>Προϋποθέσεις:</u>
Το δοχείο Β δεν είναι γεμάτο (Y≠2)
<u>Αποτέλεσμα:</u>
Το δοχείο Α είναι γεμάτο (Y=2)

T_4: Άδειασε το δοχείο Β
<u>Προϋποθέσεις:</u>
Το δοχείο Β δεν είναι άδειο (Y≠0)
<u>Αποτέλεσμα:</u>
Το δοχείο Β είναι άδειο (Y=0)

T_6: Άδειασε το δοχείο Β στο δοχείο Α
<u>Προϋποθέσεις:</u>
(1) Το δοχείο Β δεν είναι άδειο (Y≠0)
(2) Το δοχείο Α δεν είναι γεμάτο (X≠3)
<u>Αποτέλεσμα:</u>
Αδειάζουμε (όσο χωράει) από το Β στο Α
Αν |X+Y|≤3 τότε νέα κατάσταση: (X+Y,0)
Αν |X+Y|>3 τότε νέα κατάσταση: (3,Y-(3-X))

<u>Συνάρτηση Πραγματικού Κόστους:</u> Ορίζουμε ότι το **κόστος κάθε ακμής** είναι ίσο με 1 (ισοδύναμα το κόστος εφαρμογής των τελεστών μετάβασης είναι ίσο με 1).
g(n): Άθροισμα βαρών των ακμών από την αφετηρία έως τον κόμβο n.

<u>Ευρετική Συνάρτηση:</u> Ορίζουμε ως ευρετική συνάρτηση, το άθροισμα των απολύτων διαφορών των λίτρων των δοχείων σε σχέση με την κατάσταση στόχο: f(X,Y)=|X-0|+|Y-1|
Παρατήρηση: Η ευρετική είναι **ΔΕΝ είναι παραδεκτή**



f(3,0)=
|3-0|+|0-1|=4

Γράφος Καταστάσεων: