**ΟΝΟΜΑΤΕΠΩΝΥΜΟ**: ΑΘΑΝΑΣΙΟΥ ΙΩΑΝΝΗΣ

**ΑΜ**: 03117041

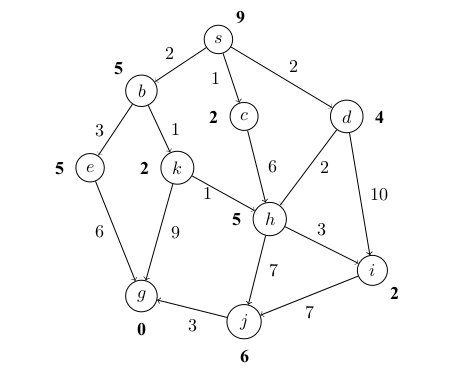
**ΕΞΑΜΗΝΟ**: 9ο

**ΗΜΕΡΟΜΗΝΙΑ**: 25/10/2021

ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ - ΧΕΙΜΕΡΙΝΟ ΕΞΑΜΗΝΟ 2021 - ΣΕΙΡΑ ΑΣΚΗΣΕΩΝ 1

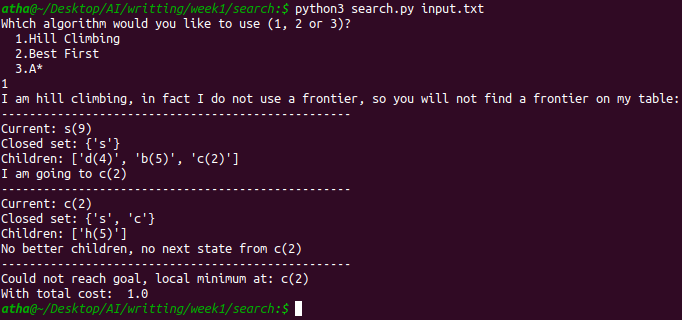
**ΑΣΚΗΣΗ 1**

Για την άσκηση αυτή, έγραψα κώδικα σε Python για εξάσκηση και για να επαληθεύσω τα αποτελέσματα στα οποία κατέληξα εκτελώντας τον αλγόριθμο με το χέρι. Ο κώδικας αυτός παρατίθεται στο τέλος του αρχείου.



1. ***Hill climbing algorithm***

* Αποτελέσματα εκτέλεσης Python:



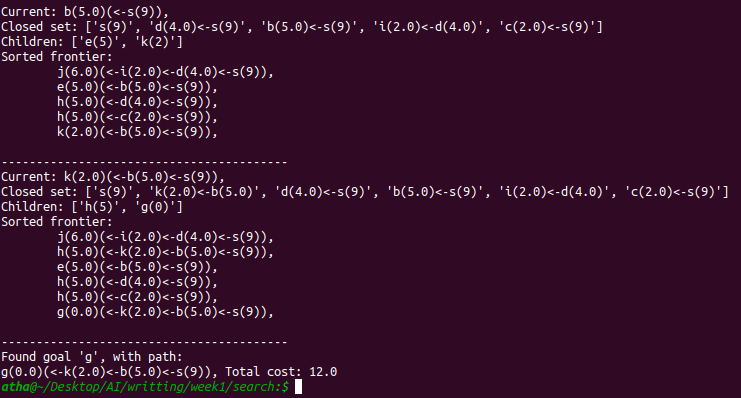
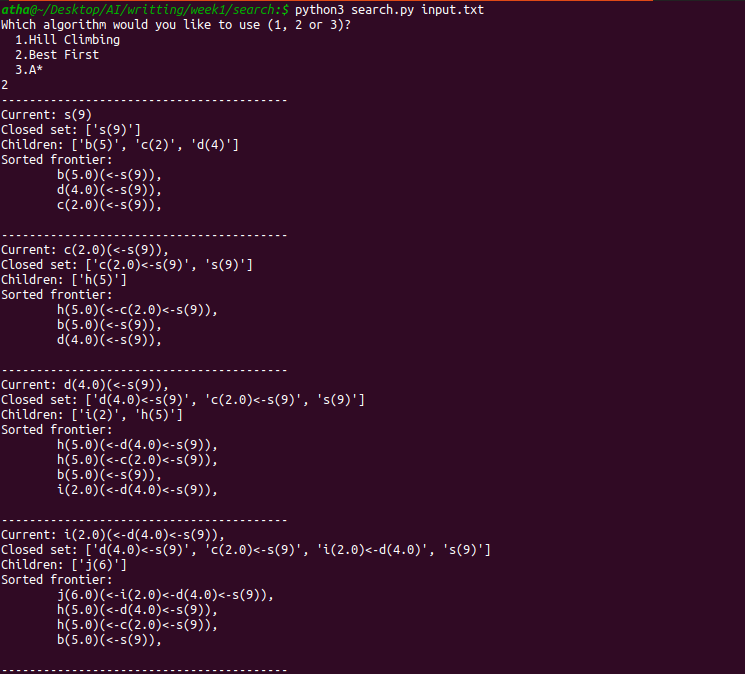
* Εκτέλεση με το χέρι:

|  |  |  |
| --- | --- | --- |
| **current state** | **closed set** | **children** |
| s(9) | s(9) | c(2), b(5), d(4) |
| c(2) | s(9), c(2) | h(5) |

* Όπως φαίνεται και από την εκτέλεση του αλγορίθμου, το c(2) αποτελεί ένα τοπικό ελάχιστο της συνάρτησης κόστους, αφού ο κόμβος c(2), έχει ως μοναδικό κόμβο “παιδί” του τον κόμβο h(5). Επομένως, ο αλγόριθμος hill-climbing δεν θα προχωρήσει σε επόμενη κατάσταση και θα τερματίσει χωρίς να βρει την λύση ‘g’.

1. ***Best-first algorithm***

* Αποτελέσματα εκτέλεσης Python:



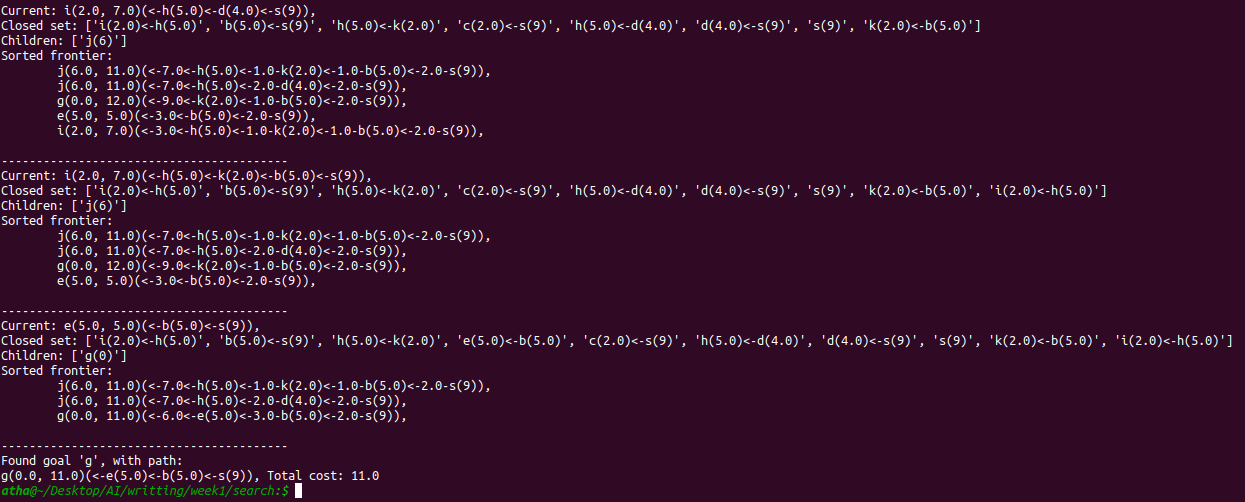
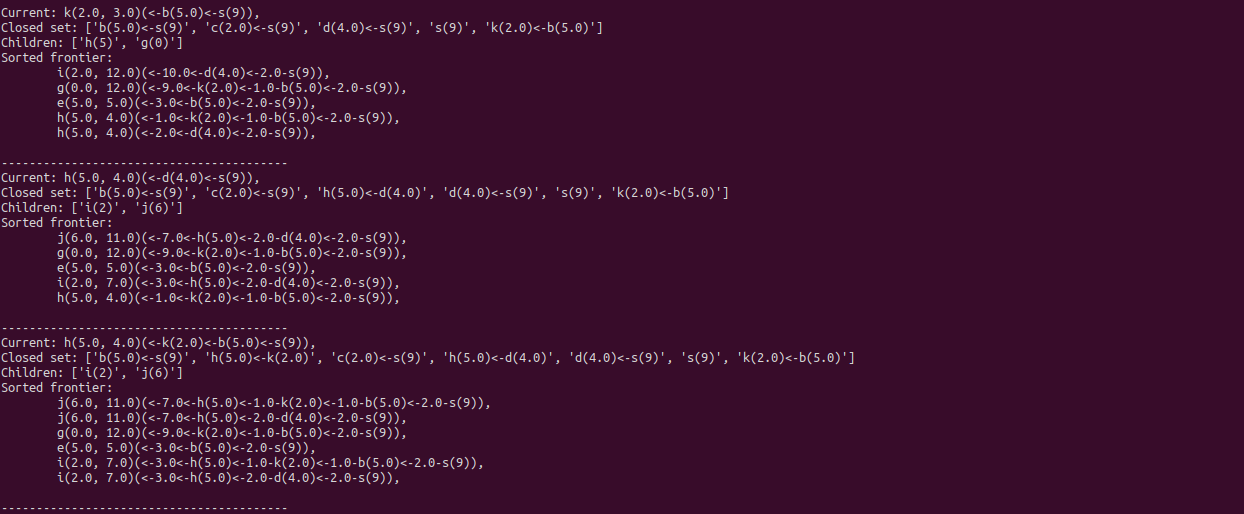
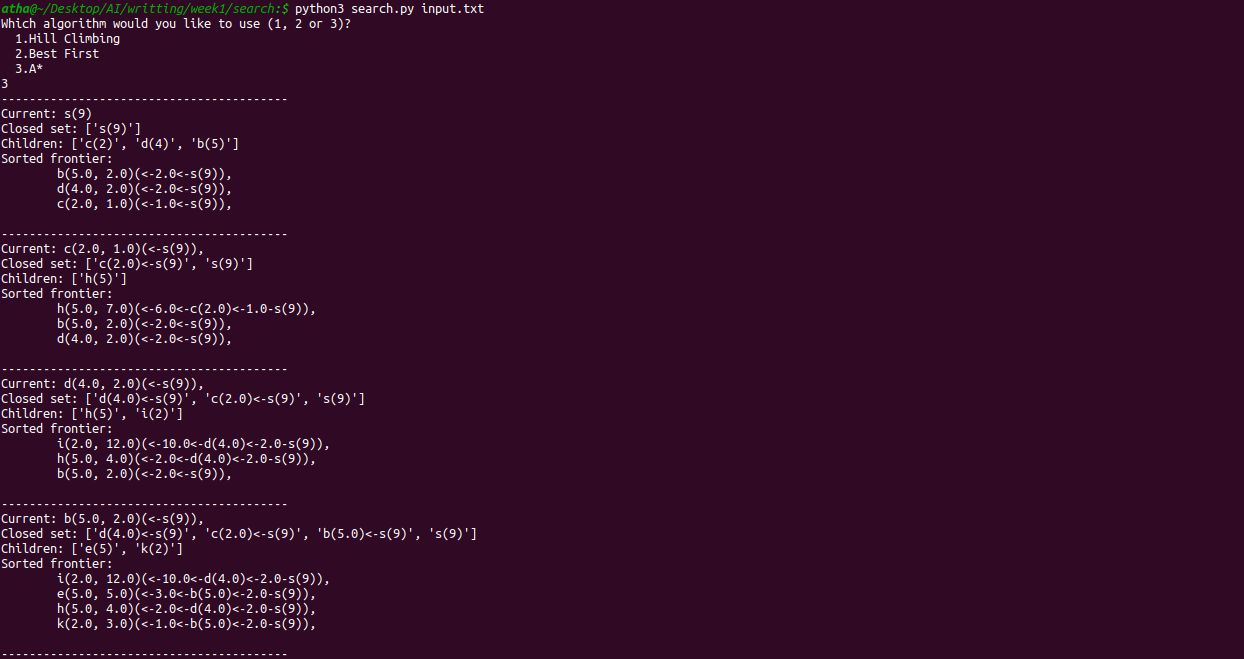
* Εκτέλεση με το χέρι:

|  |  |  |  |
| --- | --- | --- | --- |
| **current state** | **closed set** | **children** | **sorted frontier** |
| s(9) | (s, 9) | (b, 5)b<-s  (d, 4)d<-s  (c, 2)c<-s | (b, 5)b<-s  (d, 4)d<-s  (c, 2)c<-s |
| (c, 2)c<-s | (s, 9)  (c, 2)c<-s | (h, 5) h<-c<-s | (h, 5) h<-c<-s  (b, 5)b<-s  (d, 4)d<-s |
| (d, 4)d<-s | (s, 9)  (c, 2)c<-s  (d, 4)d<-s | (h, 5)h<-d<-s  (i, 2)i<-d<-s | (h, 5)h<-d<-s  (h, 5)h<-c<-s  (b, 5)b<-s  (i, 2)i<-d<-s |
| (i, 2)i<-d<-s | (s, 9)  (c, 2)c<-s  (d, 4)d<-s  (i, 2)i<-d<-s | (j, 6)j<-i<-d<-s | (j, 6)j<-i<-d<-s  (h, 5)h<-d<-s  (h, 5)h<-c<-s  (b, 5)b<-s |
| (b, 5)b<-s | (s, 9)  (c, 2)c<-s  (d, 4)d<-s  (i, 2)i<-d<-s  (b, 5)b<-s | (e, 5)e<-b<-s  (k, 2)k<-b<-s | (j, 6)j<-i<-d<-s  (e, 5)e<-b<-s  (h, 5)h<-d<-s  (h, 5)h<-c<-s  (k, 2)k<-b<-s |
| (k, 2)k<-b<-s | (s, 9)  (c, 2)c<-s  (d, 4)d<-s  (i, 2)i<-d<-s  (b, 5)b<-s  (k, 2)k<-b<-s | (g, 0)g<-k<-b<-s  (h, 5)h<-k<-b<-s | (j, 6)j<-i<-d<-s  (h, 5)h<-k<-b<-s  (e, 5)e<-b<-s  (h, 5)h<-d<-s  (h, 5)h<-c<-s  (g, 0)g<-k<-b<-s |
| (g, 0)g<-k<-b<-s |  |  |  |

* Ο αλγόριθμος τερματίζει επιτυχώς, αλλά παρατηρώ ότι δεν βρίσκει το συντομότερο μονοπάτι (βρίσκει μονοπάτι με κόστος 12, ενώ παρατηρώ ότι υπάρχει και συντομότερο μονοπάτι με κόστος 11).

1. ***A\* algorithm***

* *Θεωρώ ότι μία κατάσταση εισάγεται στο κλειστό σύνολο συνοδευόμενη από το μονοπάτι της.*
* *Δηλαδή εάν στο κλειστό σύνολο έχω τον κόμβο ‘u’ μέσω του μονοπατιού ‘p1’, και στην συνέχεια εξάγω από το μέτωπο αναζήτησης τον κόμβο ‘u’ μέσω του μονοπατιού ‘p2’, τότε ο κόμβος ‘u’ θα επεκταθεί και πάλι.*
* Αποτελέσματα εκτέλεσης Python:



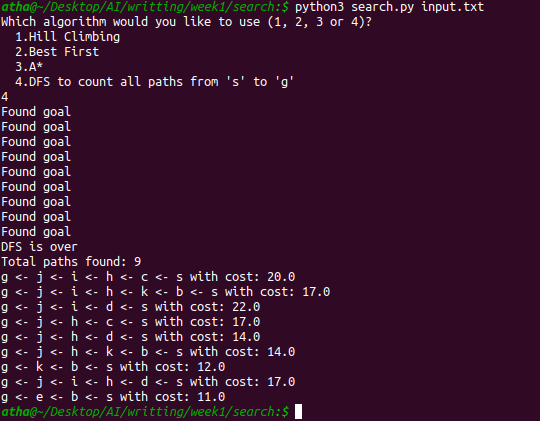
* Εκτέλεση με το χέρι:

|  |  |  |  |
| --- | --- | --- | --- |
| **current state** | **closed set** | **children** | **sorted frontier** |
| (s, 9, 0) | (s, 9, 0) | (b, 5, 2)b<-s  (c, 2, 1)c<-s  (d, 4, 2)d<-s | (b, 5, 2)b<-s  (d, 4, 2)d<-s  (c, 2, 1)c<-s |
| (c, 2, 1)c<-s | (s, 9, 0)  (c, 2, 1)c<-s | (h, 5, 7)h<-c<-s | (h, 5, 7)h<-c<-s  (b, 5, 2)b<-s  (d, 4, 2)d<-s |
| (d, 4, 2)d<-s | (s, 9, 0)  (c, 2, 1)c<-s  (d, 4, 2)d<-s | (h, 5, 4)h<-d<-s  (i, 2, 12)i<-d<-s | (i, 2, 12)i<-d<-s  (h, 5, 4)h<-d<-s  (b, 5, 2)b<-s |
| (b, 5, 2)b<-s | (s, 9, 0)  (c, 2, 1)c<-s  (d, 4, 2)d<-s  (b, 5, 2)b<-s | (e, 5, 5)e<-b<-s  (k, 2, 3)k<-b<-s | (i, 2, 12)i<-d<-s  (e, 5, 5)e<-b<-s  (h, 5, 4)h<-d<-s  (k, 2, 3)k<-b<-s |
| (k, 2, 3)k<-b<-s | (s, 9, 0)  (c, 2, 1)c<-s  (d, 4, 2)d<-s  (b, 5, 2)b<-s  (k, 2, 3)k<-b<-s | (g, 0, 12)g<-k<-b<-s  (h, 5, 4)h<-k<-b<-s | (i, 2, 12)i<-d<-s  (g, 0, 12)g<-k<-b<-s  (e, 5, 5)e<-b<-s  (h, 5, 4)h<-k<-b<-s  (h, 5, 4)h<-d<-s |
| (h, 5, 4)h<-d<-s | (s, 9, 0)  (c, 2, 1)c<-s  (d, 4, 2)d<-s  (b, 5, 2)b<-s  (k, 2, 3)k<-b<-s  (h, 5, 4)h<-d<-s | (i, 2, 7)i<-h<-d<-s  (j, 6, 11)j<-h<-d<-s | (j, 6, 11)j<-h<-d<-s  (g, 0, 12)g<-k<-b<-s  (e, 5, 5)e<-b<-s  (i, 2, 7)i<-h<-d<-s  (h, 5, 4)h<-k<-b<-s |
| (h, 5, 4)h<-k<-b<-s | (s, 9, 0)  (c, 2, 1)c<-s  (d, 4, 2)d<-s  (b, 5, 2)b<-s  (k, 2, 3)k<-b<-s  (h, 5, 4)h<-d<-s  (h, 5, 4)h<-k<-b<-s | (i, 2, 7)i<-h<-k<-b<-s  (j, 6, 11)j<-h<-k<-b<-s | (j, 6, 11)j<-h<-k<-b<-s  (j, 6, 11)j<-h<-d<-s  (g, 0, 12)g<-k<-b<-s  (e, 5, 5)e<-b<-s  (i, 2, 7)i<-h<-k<-b<-s  (i, 2, 7)i<-h<-d<-s |
| (i, 2, 7)i<-h<-d<-s | (s, 9, 0)  (c, 2, 1)c<-s  (d, 4, 2)d<-s  (b, 5, 2)b<-s  (k, 2, 3)k<-b<-s  (h, 5, 4)h<-d<-s  (h, 5, 4)h<-k<-b<-s  (i, 2, 7)i<-h<-d<-s | (j, 6, 14)j<-i<-h<-d<-s | (j, 6, 11)j<-h<-k<-b<-s  (j, 6, 11)j<-h<-d<-s  (g, 0, 12)g<-k<-b<-s  (e, 5, 5)e<-b<-s  (i, 2, 7)i<-h<-k<-b<-s |
| (i, 2, 7)i<-h<-k<-b<-s | (s, 9, 0)  (c, 2, 1)c<-s  (d, 4, 2)d<-s  (b, 5, 2)b<-s  (k, 2, 3)k<-b<-s  (h, 5, 4)h<-d<-s  (h, 5, 4)h<-k<-b<-s  (i, 2, 7)i<-h<-d<-s  (i, 2, 7)i<-h<-k<-b<-s | (j, 6, 14)j<-i<-h<-k<-b<-s | (j, 6, 11)j<-h<-k<-b<-s  (j, 6, 11)j<-h<-d<-s  (g, 0, 12)g<-k<-b<-s  (e, 5, 5)e<-b<-s |
| (e, 5, 5)e<-b<-s | (s, 9, 0)  (c, 2, 1)c<-s  (d, 4, 2)d<-s  (b, 5, 2)b<-s  (k, 2, 3)k<-b<-s  (h, 5, 4)h<-d<-s  (h, 5, 4)h<-k<-b<-s  (i, 2, 7)i<-h<-d<-s  (i, 2, 7)i<-h<-k<-b<-s  (e, 5, 5)e<-b<-s | (g, 0, 11)g<-e<-b<-s | (j, 6, 11)j<-h<-k<-b<-s  (j, 6, 11)j<-h<-d<-s  (g, 0, 11)g<-e<-b<-s |
| (g, 0, 11)g<-e<-b<-s |  |  |  |

* Παρατηρώ δύο διαδοχικές επεκτάσεις των κόμβων ‘i’ και ‘h’ καθώς κάθε φορά προέρχονταν από διαφορετικά μονοπάτια.
* Παρατηρώ ότι ο αλγόριθμος τερματίζει επιτυχώς, βρίσκοντας μονοπάτι μήκους 11 (στο συγκεριμένο παράδειγμα αυτό φαίνεται να είναι το συντομότερο μονοπάτι), άρα στο συγκεκριμένο τουλάχιστον παράδειγμα είναι βέλτιστος.

1. Για να βρούμε όλα τα μονοπάτια από τον κόμβο ‘s’ στον κόμβο ‘g’, μπορούμε να εκτελέσουμε DFS, κρατώντας τον πατέρα κάθε κόμβου και κάθε φορά που βρίσκουμε ένα νεό μονοπάτι να το κρατάμε.

Εκτελώντας DFS με το χέρι μετράω 9 μονοπάτια (9 λύσεις του προβλήματος) και για επαλήθευση εκτελώ και αντίστοιχο τμήμα κώδικα Python, κι έχω την έξοδο:



* Όπως έχω αναφέρει ήδη, η βέλτιστη λύση βρέθηκε από τον αλγόριθμο Α\* και είναι το τελευταίο μονοπάτι της εξόδου.
* Γενικά ο αλγόριθμος Α\* βρίσκει την βέλτιστη λύση όταν ο ευρετικός μηχανισμός είναι αποδεκτός (admissible) δηλαδή η τιμή της ευρετικής είναι μικρότερη ή ίση με την πραγματική τιμή.
* Στο συγκεκριμένο παράδειγμα, παρατηρώ ότι αυτό δεν συμβαίνει σε όλους τους κόμβους, αφού για ο κόμβος ‘j’ έχει ευρετική τιμή 6 και πραγματική τιμή 3.
* Επομένως, δεν θα μπορούσαμε εκ των προτέρων να είμαστε σίγουροι ότι ο Α\* θα έβρισκε βέλτιστη λύση.