Τεχνητή Νοημοσύνη

$1^η$ ομάδα ασκήσεων

Νασοπούλου Ελένη - 03121087

Άσκηση 1

Ερώτημα 1

➔ Hill Climbing

	Μέτωπο αναζήτησης	Κλειστό σύνολο	ΤΚ	Παιδιά
1	(s, 10)	{}	s	(b, 5), (c, 6), (d, 4), (a, 2)
2	$(a, 2)^{sa}$	{s}	a	(d, 4), (i, 2)
3	$(i, 2)^{sai}$	{s, a}	i	(j, 6)

=> αποτυχία

Βήμα 1
Ο τρέχον κόμβος είναι ο s(10), ο οποίος δεν είναι ο κόμβος στόχος.
Βρίσκουμε τα παιδιά του και ξεχωρίζουμε το παιδί με την μικρότερη ευρετική, δηλαδή το a(2).
Ο κόμβος a δεν έχει μεγαλύτερη ευρετική από τον s (2<10), άρα τον ορίζουμε ως τρέχον κόμβο.

Βήμα 2
Ο τρέχον κόμβος είναι ο a(2), ο οποίος δεν είναι ο κόμβος στόχος.
Βρίσκουμε τα παιδιά του και ξεχωρίζουμε το παιδί με την μικρότερη ευρετική, δηλαδή το i(2).
Ο κόμβος i δεν έχει μεγαλύτερη ευρετική από τον a (2=2), άρα τον ορίζουμε ως τρέχον κόμβο.

Βήμα 3
Ο τρέχον κόμβος είναι ο i(2), ο οποίος δεν είναι ο κόμβος στόχος.
Βρίσκουμε τα παιδιά του και ξεχωρίζουμε το παιδί με την μικρότερη ευρετική, δηλαδή το j(6).
Ο κόμβος j έχει μεγαλύτερη ευρετική από τον i (6>2), άρα ανακοινώνουμε αποτυχία.

➔ Best First Search

Περίπτωση 1
οι κόμβοι e και k μπαίνουν στο μέτωπο αναζήτησης με την εξής σειρά: (e, 2), (k, 2)

	Μέτωπο αναζήτησης	ΚΣ	ΤΚ	Παιδιά
1	$(s, 10)^{s}$	{}	s	(b, 5), (c, 6), (d, 4), (a, 2)
2	$(a, 2)^{sa}$, $(d, 4)^{sd}$, $(b, 5)^{sb}$, $(c, 6)^{sc}$	{s}	a	(d, 4), (i, 2)
3	$(i, 2)^{sai}$, $(d, 4)^{sd}$, $(b, 5)^{sb}$, $(c, 6)^{sc}$	{s, a}	i	(j, 6)
4	$(d, 4)^{sd}$, $(b, 5)^{sb}$, $(c, 6)^{sc}$, $(j, 6)^{saij}$	{s, a, i}	d	(h, 7)
5	$(b, 5)^{sb}$, $(c, 6)^{sc}$, $(j, 6)^{saij}$, $(h, 7)^{sdh}$	{s, a, i, d}	b	(e, 2), (k, 2)

6	(e, 2)^{sbe}, (k, 2)^{sbk}, (c, 6)^{sc}, (j, 6)^{saij}, (h, 7)^{sdh}	{s, a, i, d, b}	e	(g, 0)
7	(g, 0)^{sbeg}, (k, 2)^{sbk}, (c, 6)^{sc}, (j, 6)^{saij}, (h, 7)^{sdh}	{s, a, i, d, b, e}	g	-

=> επιτυχία, με μονοπάτι sbeg με κόστος 11.

<u>Περίπτωση 2</u>
οι κόμβοι e και k μπαίνουν στο μέτωπο αναζήτησης με την εξής σειρά: (k, 2), (e, 2)

	Μέτωπο αναζήτησης	**ΚΣ**	**ΤΚ**	**Παιδιά**
1	(s, 10)^{s}	{}	s	(b, 5), (c, 6), (d, 4), (a, 2)
2	(a, 2)^{sa}, (d, 4)^{sd}, (b, 5)^{sb}, (c, 6)^{sc}	{s}	a	(d, 4), (i, 2)
3	(i, 2)^{sai}, (d, 4)^{sd}, (b, 5)^{sb}, (c, 6)^{sc}	{s, a}	i	(j, 6)
4	(d, 4)^{sd}, (b, 5)^{sb}, (c, 6)^{sc}, (j, 6)^{saij}	{s, a, i}	d	(h, 7)
5	(b, 5)^{sb}, (c, 6)^{sc}, (j, 6)^{saij}, (h, 7)^{sdh}	{s, a, i, d}	b	(e, 2), (k, 2)
6	(k, 2)^{sbk}, (e, 2)^{sbe}, (c, 6)^{sc}, (j, 6)^{saij}, (h, 7)^{sdh}	{s, a, i, d, b}	k	(g, 0), (h, 7)
7	(g, 0)^{sbkg}, (e, 2)^{sbe}, (c, 6)^{sc}, (j, 6)^{saij}, (h, 7)^{sdh}	{s, a, i, d, b, k}	g	-

=> επιτυχία, με μονοπάτι sbkg με κόστος 12.

➔ Αλγόριθμος Α*

	Μέτωπο αναζήτησης	**ΚΣ**	**ΤΚ**	**Παιδιά**
1	(s, 0;10)^{s}	{}	s	(b, 2;7), (c, 1;7), (d, 2;6), (a, 1;3)
2	(a, 1;3)^{sa} , (d, 2;6)^{sd}, (b, 2;7)^{sb}, (c, 1;7)^{sc}	{(s, 0)}	a	(d, 3;7), (i, 7;9)
3	(d, 2;6)^{sd}, (b, 2;7)^{sb}, (c, 1;7)^{sc}, (i, 7;9)^{sai}	{(s, 0), (a, 1)}	d	(h, 4;11), (i, 8;10)
4	(b, 2;7)^{sb}, (c, 1;7)^{sc}, (i, 7;9)^{sai}, (h, 4;11)^{sdh}	{(s, 0), (a, 1), (d, 2)}	b	(e, 5;7), (k, 3;5)
5	(k, 3;5)^{sbk} , (c, 1;7)^{sc}, (e, 5;7)^{sbe}, (i, 7;9)^{sai}, (h, 4;11)^{sd}	{(s, 0), (a, 1), (d, 2), (b, 2)}	k	(g, 12;12), (h, 4;11)
6	(c, 1;7)^{sc}, (e, 5;7)^{sbe}, (i, 7;9)^{sai}, (h, 4;11)^{sd}, (g, 12;12)^{sbkg}	{(s, 0), (a, 1), (d, 2), (b, 2), (k, 3)}	c	(k, 2;4), (h, 5;12), (d, 2;6)
7	(k, 2;4)^{sck} , (e, 5;7)^{sbe}, (i, 7;9)^{sai}, (h, 4;11)^{sd}, (g, 12;12)^{sbkg}	{(s, 0), (a, 1), (d, 2), (b, 2), (c, 1)}	k	(g, 11;11), (h, 3;10)
8	(e, 5;7)^{sbe}, (i, 7;9)^{sai}, (h, 3;10)^{sckh}, (g, 11;11)^{sckg}	{(s, 0), (a, 1), (d, 2), (b, 2), (c, 1), (k, 2)}	e	(g, 11;11)
9	(i, 7;9)^{sai}, (h, 3;10)^{sckh}, (g, 11;11)^{sckg}	{(s, 0), (a, 1), (d, 2), (b, 2), (c, 1), (k, 2), (e, 5)}	i	(j, 14;20)

10	(h, 3;10)^{sckh}, (g, 11;11)^{sckg}, (j, 14;20)^{saij}	{(s, 0), (a, 1), (d, 2), (b, 2), (c, 1), (k, 2), (e, 5), (i, 7)}	h	(i, 6;8), (j, 14;20), (g, 12;12)
11	(i, 6;8)^{sckhi} , (g, 11;11)^{sckg}, (j, 14;20)^{saij}	{(s, 0), (a, 1), (d, 2), (b, 2), (c, 1), (k, 2), (e, 5), (h, 3)}	i	(j, 10;12)
12	**(g, 11;11)^{sckg}**, (j, 10;12)^{sckhij}	{(s, 0), (a, 1), (d, 2), (b, 2), (c, 1), (k, 2), (e, 5), (h, 3), (i, 6)}	g	-
13	(j, 10;12)^{sckhij}	{(s, 0), (a, 1), (d, 2), (b, 2), (c, 1), (k, 2), (e, 5), (h, 3), (i, 6), (g, 11)}	j	(g, 13;13)
14	-	{(s, 0), (a, 1), (d, 2), (b, 2), (c, 1), (k, 2), (e, 5), (h, 3), (i, 6), (g, 11), (j, 10)}		

=> επιτυχία, με μονοπάτι sckg με κόστος 11.

Ερώτημα 2

Όλες οι λύσεις του προβλήματος φαίνονται στον παρακάτω πίνακα.

	Μονοπάτι	**Κόστος**
1	sbeg	11
2	sbkg	12
3	sbkhg	13
4	sbkhjg	14
5	sbkhijg	17
6	sckg	11
7	sckhg	12
8	sckhjg	13
9	sckhijg	16
10	schg	14
11	schjg	15

12	schijg	18
13	scdhg	13
14	scdhjg	14
15	scdhijg	17
16	scdijg	18
17	sdhg	13
18	sdhjg	14
19	sdhijg	17
20	sdijg	18
21	sadhg	14
22	sadhjg	15
23	sadhijg	18
24	sadijg	19
25	saijg	17

Συνεπώς, το πρόβλημα έχει 25 λύσεις, από τις οποίες οι 2 είναι βέλτιστες. Πιο συγκεκριμένα, οι βέλτιστες λύσεις είναι:

- sbeg (κόστος 11)
- sckg (κόστος 11)

Οι αλγόριθμοι δίνουν τις παρακάτω λύσεις. Αξίζει να σημειωθεί πως η σειρά που θα μπουν οι κόμβοι στο μέτωπο αναζήτησης στους αλγορίθμους Best First και A* στην περίπτωση ισοβαθμίας οδηγεί σε δύο διαφορετικά σενάρια λύσης που δίνουν διαφορετικά μονοπάτια.

Hill Climbing: αποτυχία – δεν βρίσκει λύση
Best First Search:

- sbeg: βέλτιστη – κόστος 11
- sbkg: όχι βέλτιστη – κόστος 12

Αλγόριθμος A*:

- sckg: βέλτιστη – κόστος 11

Όσον αφορά τους αλγόριθμους Hill Climbing και Best First Search, <u>δεν</u> μπορούμε να είμαστε σίγουροι εκ των προτέρων ότι θα βρει βέλτιστη λύση, καθώς αυτή βασίζεται αποκλειστικά στην τιμή της ευριστικής και δεν συνυπολογίζει τις πραγματικές αποστάσεις.

Όσον αφορά τον αλγόριθμο A*, <u>δεν</u> μπορούμε να είμαστε σίγουροι εκ των προτέρων ότι θα βρει βέλτιστη λύση, καθώς η ευριστική δεν είναι συνεπής. Πιο συγκεκριμένα, η πραγματική απόσταση του κόμβου g(j)=3 είναι μικρότερη από την τιμή της ευρετικής h(j)=6.

<u>Ερώτημα 3</u>

Ο αλγόριθμος A* βρίσκει συμπτωματικά την βέλτιστη λύση, παρόλο που η ευρετική δεν είναι συνεπής. Αυτό συμβαίνει διότι ο κόμβος με τη μη συνεπή τιμή, δηλαδή ο j, δεν επηρεάζει τα βέλτιστα μονοπάτια.

Η ελάχιστη δυνατή τροποποίηση που θα μπορούσαμε να κάνουμε ώστε να μην βρίσκει βέλτιστη λύση, θα ήταν να μεταβάλουμε την ευρετική των κόμβων e και k σε μη συνεπείς τιμές. Έτσι, για h(e)=11 και h(k)=11, ο αλγόριθμος A* σίγουρα δεν θα βρει τη βέλτιστη λύση

Ερώτημα 4

Η ευρετική δεν είναι συνεπής. Υπάρχει, δηλαδή, κόμβος στον οποίο η τιμή της ευρετικής είναι μεγαλύτερη από την πραγματική απόσταση του κόμβου από τον στόχο. Πιο συγκεκριμένα, η πραγματική απόσταση του κόμβου g(j)=3 είναι μικρότερη από την τιμή της ευρετικής h(j)=6. Η ελάχιστη δυνατή τροποποίηση είναι να μεταβάλουμε την ευρετική του κόμβου j σε 1, 2 ή 3.

Ερώτημα 5

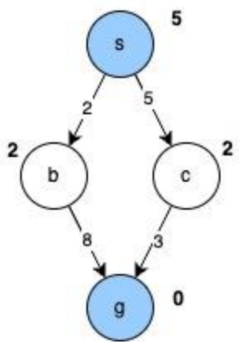
Η ακριβέστερη συνεπής ευρετική που θα μας δίνει γρήγορα τη βέλτιστη λύση είναι η πραγματική απόσταση του κάθε κόμβου από τον στόχο.

Κόμβος	**Ευρετική**
s	11
a	11
b	9
c	10
d	11
e	6

Κόμβος	**Ευρετική**
h	9
i	10
j	3
k	9
g	0

Άσκηση 2

Έστω ότι εκτελείτε ο αλγόριθμος A*, σε πρόβλημα με αποδεκτή ευρετική, μέχρι να βρει λύση για πρώτη φορά ένα μονοπάτι προς τον στόχο και μετά τερματίζει. Υπάρχει περίπτωση αυτή η λύση να μην είναι βέλτιστη, όπως συμβαίνει στο παρακάτω παράδειγμα.

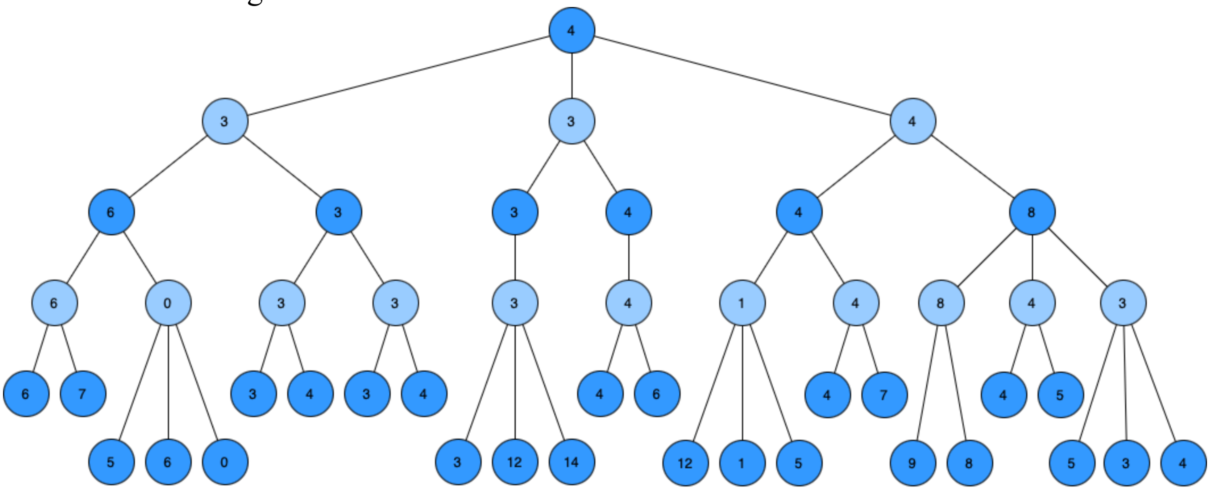


	Μέτωπο αναζήτησης	**ΚΣ**	**ΤΚ**	**Παιδιά**
1	(s, 0;5)^{s}	{}	s	(b, 2;2), (c, 5;2)
2	(b, 2;2)^{sb}, (c, 5;2)^{sc}	{s}	b	(g, 10;0)
3	(c, 5;2)^{sc}, **(g, 10;0)^{sbg}**	{s, k}	-	-

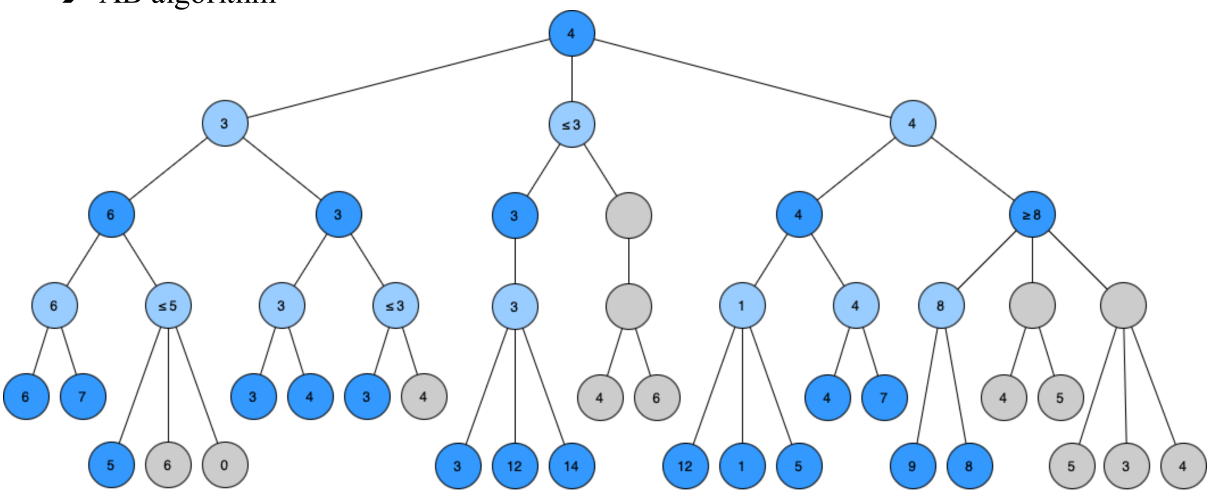
Η ευρετική είναι συνεπής, διότι η τιμή της είναι μικρότερη ή ίση από την πραγματική απόσταση του κάθε κόμβου από τον στόχο και το βέλτιστο μονοπάτι από την αφετηρία s προς τον κόμβο στόχο g είναι το scg (κόστους 8).
Ο αλγόριθμος A* εκτελείτε μέχρι να βρει μονοπάτι προς τον στόχο για πρώτη φορά, δηλαδή το sbg, το οποίο **δεν** είναι βέλτιστο, καθώς έχει κόστος 10.

Άσκηση 3

➔ Minimax algorithm



➔ AB algorithm



Σειρά επίσκεψης κόμβων:
1, 2, 5, 11, 22, 23, 12, 24, 6, 13, 27, 28, 14, 29, 3, 7, 15, 31, 32, 33, 4, 9, 17, 36, 37, 38, 18, 39, 40, 10, 19, 41, 42

Άσκηση 4

Ερώτημα 1

Μορφή χρωμοσώματος: x = abcdefgh
Συνάρτηση προσαρμοστικότητας χρωμοσώματος: $f(x) = (a + b + c + d) - (e + f + g + h)$

Αρχική γενιά:

Άτομο	Χρωμόσωμα	Fitness Function	Fitness
1	x_1 = 98765432	$f(x_1) = (9+8+7+6) - (5+4+3+2)$	16
2	x_2 = 43212121	$f(x_2) = (4+3+2+1) - (2+1+2+1)$	4
3	x_3 = 91827364	$f(x_3) = (9+1+8+2) - (7+3+6+4)$	0
4	x_4 = 20000001	$f(x_4) = (2+0+0+0) - (0+0+0+1)$	1

Συνολική προσαρμοστικότητα/ απόδοση: 21

Φθίνουσα σειρά προσαρμοστικότητας: $x_1 > x_2 > x_4 > x_3$

Ερώτημα 2

- ➔ Crossover x_1, x_2 στο μέσο του χρωμοσώματος

Γ1 = (9 8 7 6 | 5 4 3 2) => Π1 = (9 8 7 6 2 1 2 1)
Γ2 = (4 3 2 1 | 2 1 2 1) => Π2 = (4 3 2 1 5 4 3 2)

- ➔ Crossover x_2, x_4 μεταξύ των σημείων b και f

Γ1 = (4 3 | 2 1 2 1 | 2 1) => Π1 = (4 3 0 0 0 0 2 1)
Γ2 = (2 0 | 0 0 0 0 | 0 1) => Π2 = (2 0 2 1 2 1 0 1)

- ➔ Crossover x_1, x_3 μεταξύ των τριών τελευταίων γονιδίων

Γ1 = (9 8 7 6 5 | 4 3 2) => Π1 = (9 8 7 6 5 3 6 4)
Γ2 = (9 1 8 2 7 | 3 6 4) => Π2 = (9 1 8 2 7 4 3 2)

Ερώτημα 3

Επόμενη γενιά:

Άτομο	Χρωμόσωμα	Fitness Function	Fitness
1	x_1 = 98762121	$f(x_1) = (9+8+7+6) - (2+1+2+1)$	24
2	x_2 = 43215432	$f(x_2) = (4+3+2+1) - (5+4+3+2)$	-4
3	x_3 = 43000021	$f(x_3) = (4+3+0+0) - (0+0+2+1)$	4
4	x_4 = 20212101	$f(x_4) = (2+0+2+1) - (2+1+0+1)$	1
5	x_5 = 98765364	$f(x_5) = (9+8+7+6) - (5+3+6+4)$	12
6	x_6 = 91827432	$f(x_6) = (9+1+8+2) - (7+4+3+2)$	4

Συνολική προσαρμοστικότητα/ απόδοση: 41

Η συνολική προσαρμοστικότητα του νέου πληθυσμού έχει βελτιωθεί σε σχέση με αυτή του παλιού.

Ερώτημα 4

Για να υπολογίσουμε την μέγιστη προσαρμοστικότητα ενός χρωμοσώματος πρέπει:

- τα πρώτα τέσσερα ψηφία (a, b, c, d) να είναι το μέγιστο δυνατό άθροισμα (δηλαδή 9 + 9 + 9 + 9 = 36), άρα (a, b, c, d) = (9, 9, 9, 9)
- τα τελευταία τέσσερα ψηφία (e, f, g, h) είναι το ελάχιστο δυνατό άθροισμα (δηλαδή 0 + 0 + 0 + 0 = 0), άρα (e, f, g, h) = (0, 0, 0, 0)

Συνεπώς, το χρωμόσωμα με την μέγιστη αποδοτικότητα είναι το x = 99990000 με αποδοτικότητα f(x) = (9+9+9+9) – (0+0+0+0) = 36.