

Εργασία τεχνητής νοημοσύνης

Ιωακείμ ΕΛ Χαττάμπ-Μπριστογιάννης Π19048

Πίνακας περιεχομένων

Αρχικοποίηση πληθυσμού	1
Εύρεση καταλληλότητας	1
Κεντρικός βρόγχος.....	2
Μερική ανανέωση πληθυσμού	2
Αναπαραγωγή	2
Μετάλλαξη	2
Εκτελέσιμο.....	2
Παραδείγματα εκτέλεσης	3

Αρχικοποίηση πληθυσμού

Η αρχικοποίηση του πληθυσμού γίνεται με την βοήθεια της συνάρτησης `InitializePopulation`, όπου στην οποία, αρχικοποιούμε την κενή λίστα `population` και στην συνέχεια για όσο έχει ζητήσει ο χρήστης να είναι το μέγεθος του πληθυσμού ,δημιουργούμε ένα κενό χρωμόσωμα όπου το γεμίζουμε με τις πόλεις με τυχαία σειρά, ενώ ελέγχουμε ταυτόχρονα να μην υπάρχουν διπλότυπες πόλεις και προσθέτουμε το χρωμόσωμα αυτό στην λίστα `population`. Τελικά η συνάρτηση επιστρέφει την λίστα `population` με τυχαία αρχικοποιημένες πόλεις.

Εύρεση καταλληλότητας

Το επόμενο βήμα είναι να υπολογίσουμε την καταλληλότητα της κάθε λύσης στο πληθυσμό που αρχικοποιήσαμε. Αυτό το κάνουμε για κάθε μία με την βοήθεια της συνάρτησης `Suitability`, η οποία υπολογίζει το κόστος ολοκλήρης της διαδρομής, το αντιστρέφει και το επιστέφει.

Κεντρικός βρόγχος

Στην συνέχεια προχωράμε στον κεντρικό κόμβο του αλγόριθμου, οι συνθήκες τερματισμού σε αυτή την υλοποίηση είναι είτε η καλύτερη λύση έως τώρα να έχει μεγαλύτερη καταλληλότητα από αυτή που έχουμε ορίσει, είτε να έχουμε φτάσει στις χίλιες γενεές.

Μερική ανανέωση πληθυσμού

Πρώτο πράγμα που κάνουμε στον κεντρικό βρόγχο είναι να επιλέξουμε μερικές λύσεις οι οποίες θα μεταφερθούν στον νέο πληθυσμό αυτούσιες. Σε αυτή τη υλοποίηση συγκρατούμε το 30% του πληθυσμού και η επιλογή των λύσεων γίνεται με την τεχνική της ρουλέτας, καθώς και ελέγχουμε να μην επιλεγούν διπλότυπες λύσεις.

Αναπαραγωγή

Επόμενο βήμα είναι η αναπαραγωγή του υπολειπόμενου πληθυσμού. Επιλέγουμε του γονείς με την τεχνική της ρουλέτας και στην συνέχεια δημιουργούμε τυχαία ζευγάρια μεταξύ τους. Για κάθε ένα ζευγάρι γονέων, δημιουργούμε δύο παιδιά με Διασταύρωση ενός σημείου, κρατώντας το πρώτο μισό του πρώτου γονέα και το δεύτερο μισό του δεύτερου γονέα για το πρώτο παιδί και αντιστρόφως για το δεύτερο παιδί.

Μετάλλαξη

Τελευταίο βήμα του κεντρικού βρόγχου είναι η μετάλλαξη. Επιλέγουμε τυχαία το 10% του νέου πληθυσμού, για κάθε μία από τις επιλεγόμενες λύσεις επιλέγουμε τυχαία δύο πόλεις και τους αλλάζουμε θέσεις μεταξύ τους.

Τελικά αφού ολοκληρωθεί η εκτέλεση του κεντρικού κόμβου επιστρέφουμε την μεγαλύτερη τιμή της συνάρτησης καταλληλότητας, την λύση με την μεγαλύτερη τιμή καταλληλότητας καθώς και τον αριθμό γενεών.

Εκτελέσιμο

Στο εκτελέσιμο κομμάτι του κώδικα αρχικά, ζητάμε από τον χρήστη να μας δώσει τον αριθμό των πόλεων που επιθυμεί να έχει το TSP πρόβλημα. Στην συνέχεια με αυτή την πληροφορία δημιουργούμε με την βοήθεια της μεθόδου `CreateAdjacencyMatrix` τον πίνακα γειτνίασης του προβλήματος και καλούμε τον αλγόριθμο που περιγράψαμε παραπάνω για να το επιλύσουμε. Τελικά προβάλουμε στην κονσόλα τον αριθμό γενεών που χρειάστηκαν για να βρεθεί λύση, το κόστος της λύσης, καθώς και την σειρά των πόλεων που αποτελούν την βέλτιστη λύση την μία κάτω από την άλλη.

Παραδείγματα εκτέλεσης

Παρακάτω δίνονται παραδείγματα εκτέλεσης του προγράμματος :

Microsoft Visual Studio Debug Console

```
Enter the ammount of cities:
78
the number of generations was :7
the solutions costs is :305
The solution is
37
↓
71
↓
61
↓
47
↓
19
↓
74
↓
59
↓
62
↓
30
↓
21
↓
39
↓
11
↓
53
↓
77
↓
26
↓
66
↓
20
↓
5
↓
42
↓
8
↓
40
↓
22
↓
34
```

Microsoft Visual Studio Debug Console

```
Enter the amount of cities:
100
the number of generations was :1000
the solutions costs is :434
The solution is
87
4
0
4
82
4
45
4
89
4
74
4
91
4
67
4
47
4
19
4
53
4
75
4
93
4
88
4
7
4
17
4
48
4
36
4
79
4
8
4
26
4
52
4
95
```

Microsoft Visual Studio Debug Console

```
Enter the amount of cities:
22
the number of generations was :0
the solutions costs is :78
The solution is
19
4
4
4
8
4
2
4
15
4
14
4
12
4
3
4
8
4
21
4
16
4
18
4
13
4
6
4
11
4
5
4
18
4
9
4
7
4
20
4
17
4
1
4
19
```

Microsoft Visual Studio Debug Console

```
Enter the amount of cities:
97
the number of generations was :15
the solutions costs is :378,99997
The solution is
78
↓
71
↓
96
↓
55
↓
32
↓
45
↓
24
↓
73
↓
75
↓
89
↓
65
↓
7
↓
41
↓
18
↓
39
↓
0
↓
82
↓
31
↓
0
↓
61
↓
92
↓
67
↓
30
```

Microsoft Visual Studio Debug Console

```
Enter the amount of cities:
999
the number of generations was :1000
the solutions costs is :5013
The solution is
919
↓
803
↓
917
↓
27
↓
708
↓
588
↓
90
↓
864
↓
923
↓
984
↓
260
↓
531
↓
981
↓
915
↓
886
↓
273
↓
145
↓
411
↓
259
↓
884
↓
844
↓
7
↓
655
```