

1^Η ΕΡΓΑΣΙΑ ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ

ΑΘΑΝΑΣΙΟΣ ΚΛΕΤΤΑΣ : 3180079

ΙΑΣΩΝ ΖΙΩΓΑΣ : 3180057

- 1) Για το πρόβλημα των κανίβαλων και των ιεραποστόλων έχουμε μια κλάση STATE η οποία αντιπροσωπεύει μια κατάσταση στο πρόβλημα μας. Η STATE έχει σαν attributes τον αριθμό των κανίβαλων που είναι ίσος με τον αριθμό των ιεραποστόλων(N) και το μέγεθος της βάρκας(M). Στην STATE υπάρχει η μέθοδος getChildren() η οποία παράγει ένα σύνολο καταστάσεων (“παιδιών”) με βάση την τρέχουσα κατάσταση χρησιμοποιώντας ένα σύνολο κανόνων οι οποίοι ορίζονται από την μέθοδο moveBoat(). Η moveBoat() ελέγχει για την εκάστοτε child κατάσταση εάν η δοθείσης “μεταφορά” είναι ικανοποιήσιμη για τις μεταβλητές που του δώσαμε. Επίσης κάθε κατάσταση child έχει μια μεταβλητή που δηλώνει το ευρετικό του σκορ (heuristicScore) το οποίο υπολογίζεται από την μέθοδο calculateHeuristicScore(). Εφόσον παραχθούν όλα τα παιδιά με την βοήθεια του A* αλγόριθμου αναζητούμε και επιστρέφουμε την βέλτιστη λύση. Ο A* με την χρήση μια σειράς ελέγχει την “γρηραιότερο” κατάσταση εάν αποτελεί λύση. Εάν η κατάσταση υπάρχει στο κλειστό σύνολο του A* δεν ελέγχουμε καθόλου την κατάσταση. Αντίθετα εάν η κατάσταση αποτελεί λύση το πρόγραμμα τελειώνει αλλιώς παράγει τα παιδιά της κατάστασης και τα προσθέτει στην ουρά. Τέλος ταξινομούμε την ουρά με βάση το heuristicScore. Τέλος γίνεται μια visual αναπαράσταση κάνοντας backtracking από την τελευταία κατάσταση που είναι η λύση μέχρι την αρχική χρησιμοποιώντας την μέθοδο showState()).

Δυνατότητες :

Πρόβλημα Ιεραπόστολων και κανίβαλων	N = 50	N =70	N=100
M = 4	0.016 sec	0.024 sec	0.046 sec
M = 6	0.013 sec	0.018 sec	0.035 sec
M = 7	0.012 sec	0.019 sec	0.031 sec

(*CrossingsAllowed = 1000)

- 2) Για το πρόβλημα των **N** βασίλισσών χρησιμοποιήσαμε γενετικό αλγόριθμο στον οποίο χρησιμοποιούμε χρωμοσώματα(Chromosome). Τα χρωμοσώματα έχουν σαν μεταβλητές τα γονίδια που περιγράφουν την θέση της κάθε βασίλισσας στην στήλη της και το score, το οποίο score υπολογίζεται από την findScore() η οποία ελέγχει πόσες βασίλισσες δεν απειλούνται από τις άλλες. Επιπλέον στην Chromosome υπάρχει η μέθοδος mutate(int N) η οποία αλλάζει έναν αριθμό από τα γονίδια (δηλαδή την θέση μια βασίλισσας στην στήλη της). Το πρόγραμμα μας ξεκινάει με την κλήση της μεθόδου geneticAlgorithm(populationSize, mutationProbability,maxLoops,**N**) . Στον Genetic υπάρχει η μέθοδος geneticAlgorithm στην οποία αρχικοποιούμε το σκορ της λύσης, δημιουργούμε ένα σύνολο χρωμοσωμάτων(Chromosome) ίσο με το populationSize και μετά ενημερώνουμε το prorabilityArray χρησιμοποιώντας την μέθοδο updateprorabilityArray() στην οποία τα χρωμοσώματα με μεγαλύτερο σκορ εμφανίζονται περισσότερο. Στην συνέχεια μέσω μιας επανάληψης επιλέγουμε δυο διαφορετικά χρωμοσώματα από την prorabilityArray τα διχοτομούμε σε ένα τυχαίο σημείο και παράγουμε ένα νέο χρωμόσωμα από αυτά τα δυο μέσω της μεθόδου reproduce().Έπειτα επιλέγουμε έναν τυχαίο αριθμό από το 0 έως το 1 και εάν είναι

μικρότερος από το mutationProbability το κάνουμε mutate το νέο χρωμόσωμα.
Αυτό επαναλαμβάνεται μέχρι κάποιο χρωμόσωμα να είναι λύση ή μέχρι να
εξαντληθούν τα loops. Τέλος κάνουμε visual αναπαράσταση το χρωμόσωμα που
είναι η λύση ή το χρωμόσωμα που έφτασε πιο κοντά στην λύση

Δυνατότητες :

Πρόβλημα N Βασιλισσών	N = 5	N =8	N=12
	0.015 sec	0.2 sec	53.674 sec

(*PopulationSize = 300, mutationProbability = 8%, loops = 10000)