

Υπολογιστική Νοημοσύνη Εργαστηριακή Άσκηση Β'

Φλώρος Παναγιώτης 1047137

CEID
University of Patras
2018-2019

B1. Σχεδιασμός Γενετικού Αλγορίθμου

A. Κωδικοποίηση

Η αναπαράσταση που χρησιμοποιήθηκε για την υλοποίηση του προγράμματος είναι ένας τρισδιάστατος πίνακας με διαστάσεις (τάξη, δώρο, μέρα), του οποίου τα κελιά περιέχουν έναν ακέραιο, είτε το -1 αν δεν υπάρχει μάθημα, είτε το id του μαθήματος αν υπάρχει.

B. Διαδικασία Επιδιόρθωσης

Δεν υλοποιήθηκε διαδικασία επιδιόρθωσης, διότι όλες οι λύσεις δημιουργούνται με τέτοιο τρόπο, ώστε να μην παραβιάζουν τους κανόνες νομιμότητας. Αυτό επιτυγχάνεται με έλεγχο κατά την τυχαία τοποθέτηση κάθε στοιχείου για παραβίαση των κανόνων νομιμότητας. Στην περίπτωση κατά την οποία αυτή παραβιάζεται, το στοιχείο τοποθετείται σε άλλη τυχαία θέση.

Γ. Συνάρτηση Καταλληλότητας

Η συνάρτηση βελτιστοποίησης λαμβάνει υπόψη τους περιορισμούς βελτιστοποίησης:

1. Κάθε φορά που προσφέρονται μαθήματα του ίδιου τομέα την ίδια μέρα και ώρα, το άτομο χάνει 2 πόντους fitness.
2. Για κάθε μάθημα, το οποίο ακολουθεί κάποιο άλλο χωρίς κενό, το άτομο κερδίζει 3 πόντους fitness.
3. Για κάθε μάθημα που προσφέρεται σε μη συνεχόμενες ημέρες, το άτομο κερδίζει 10 πόντους fitness.

Λεπτομέρειες υλοποίησης:

1. Για κάθε δίκωρο και μέρα, ελέγχονται οι αίθουσες για το αν υπάρχει μάθημα. Αν υπάρχει μάθημα, αυξάνεται διάνυσμα που αποθηκεύει το πλήθος του κάθε τομέα. Τελικά, αν κάποιο πλήθος είναι διάφορο του 0 και του 1, εφαρμόζεται μείωση του fitness.
2. Για κάθε τάξη και μέρα, αποθηκεύεται διάνυσμα με τις θέσεις των μαθημάτων ως δίκωρα. Αν η διαφορά δύο διαδοχικών στοιχείων του διανύσματος είναι 1, σημαίνει ότι τα μαθήματα είναι συνεχόμενα, οπότε εφαρμόζεται αύξηση του fitness.
3. Για κάθε μάθημα το οποίο διδάσκεται πάνω από ένα δίκωρο, αποθηκεύεται η θέση του κάθε δίκωρου ως μέρα (δηλαδή αν ένα μάθημα διδάσκεται Δευτέρα και Πέμπτη, αποθηκεύεται $[0, 3]$). Αν η διαφορά των θέσεων είναι διάφορη του 1, τότε το μάθημα δεν διδάσκεται συνεχόμενες μέρες, άρα εφαρμόζεται αύξηση του fitness.

Δ. Γενετικοί Τελεστές

- **Επιλογή:** Η επιλογή υλοποιήθηκε με τη μέθοδο ρουλέτα με σχισμές (*slotted roulette wheel*).
- **Διασταύρωση:** Για αυτόν τον τελεστή χρησιμοποιήθηκε λίστα που περιείχε τα δίκωρα. Το *cross point* εφαρμόστηκε σε αυτή τη λίστα, ώστε να διατηρηθεί η συνθήκη νομιμότητας, η οποία απαιτεί να υπάρχει στο πρόγραμμα ακριβώς ο αριθμός των μαθημάτων και τα μαθήματα επιλογής με τις ώρες τους. Ούτε περισσότερα ούτε λιγότερα. Η εφαρμογή του *cross point* στον τρισδιάστατο πίνακα της αναπαράστασης είναι μια αρκετά κακή ιδέα, καθώς είναι σχετικά απίθανο να χωριστεί το πρόγραμμα με τέτοιο τρόπο, ώστε να τηρούνται όλοι οι κανόνες νομιμότητας. Συνήθως έχει ως αποτέλεσμα ιδιαίτερα κακές λύσεις. Αφού χωριστεί η λίστα δίκωρων σε 2 κομμάτια, πρέπει να σχηματιστούν τα 2 νέα άτομα του πληθυσμού και να αντικαταστήσουν τα παλιά. Για κάθε στοιχείο της λίστας, εντοπίζεται η θέση του στο τρισδιάστατο πίνακα για το πρώτο άτομο που επιλέχθηκε για διασταύρωση και τοποθετείται στο πρώτο νέο άτομο. Η διαδικασία συνεχίζεται, μέχρι το πρώτο νέο άτομο να γεμίσει με τα στοιχεία του πρώτου μισού της λίστας δίκωρων στις αντίστοιχες θέσεις από το πρώτο γονέα. Ομοίως, το δεύτερο μισό του πρώτου νέου ατόμου γεμίζει με το δεύτερο μισό της λίστας δίκωρων, αλλά αυτή τη φορά στις αντίστοιχες θέσεις από το δεύτερο γονέα, ελέγχοντας τον κανόνα νομιμότητας 3 και λύνοντας τυχόν conflict (δηλαδή το στοιχείο να 'πέσει' σε θέση που τοποθετήθηκε άλλο προηγουμένως από το πρώτο μισό της διαδικασίας). Με παρόμοιο τρόπο υπολογίζεται και το δεύτερο παιδί. Η παραπάνω διαδικασία σαφώς εκτελείται στο σύνολο των επιλεχθέντων στοιχείων για cross, το πλήθος του οποίου είναι ζυγό (όταν επιλεγεί περιττό πλήθος, τότε αφαιρείται ένα άτομο).

- **Μετάλλαξη:** Για κάθε δίκωρο που υπάρχει στο τρισδιάστατο πίνακα, αν τύχει να γίνει μετάλλαξη, επιλέγεται μία τυχαία θέση και μετακινείται εκεί το μάθημα, ελέγχοντας και τηρώντας πάντα τους κανόνες νομιμότητας.

Ε. Αρχικός πληθυσμός

Ο αρχικός πληθυσμός δημιουργείται ως ένας πίνακας με n *individuals*, ο καθένας εκ των οποίων έχει ένα τρισδιάστατο πίνακα που αρχικοποιείται με την τιμή -1 σε όλες τις θέσεις και έπειτα τα μαθήματα. Χρησιμοποιώντας τη λίστα δίκωρων, τοποθετούνται τυχαία στον πίνακα φροντίζοντας να διατηρούνται όλοι οι κανόνες νομιμότητας. Το κάθε άτομο του πληθυσμού ουσιαστικά αποτελείται από 29 δίκωρα, άρα από 29 γονίδια με το καθένα να περιέχει το id και τη θέση $[i][j][k]$.

B2. Υλοποίηση Γενετικού Αλγορίθμου

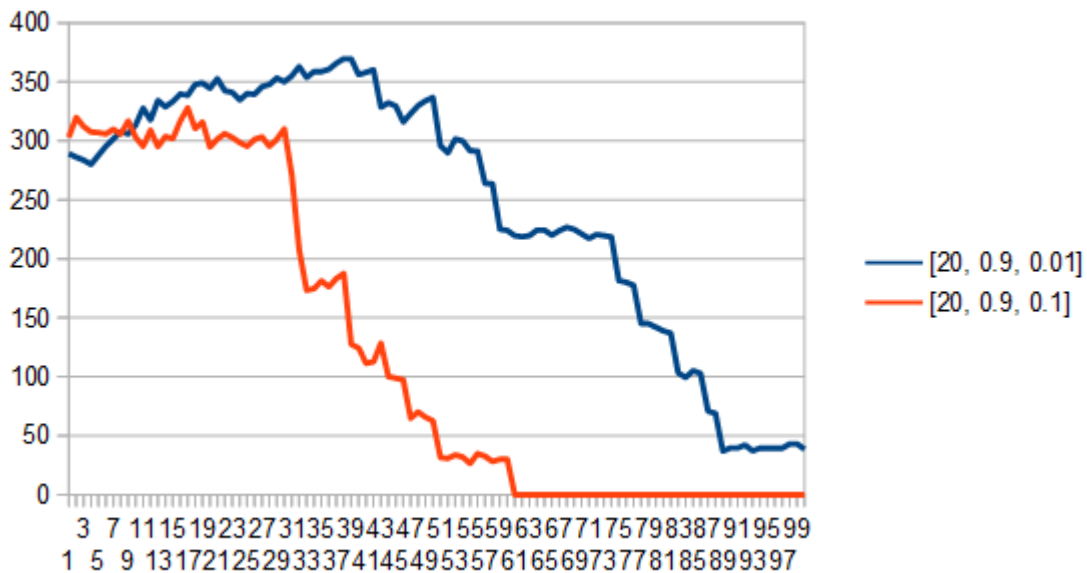
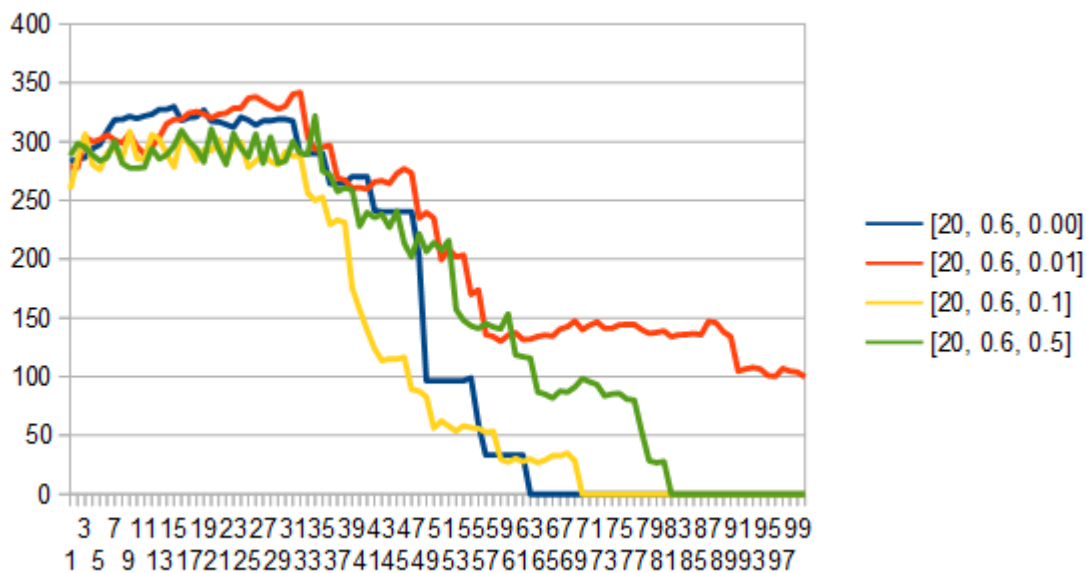
Η άσκηση υλοποιήθηκε σε python και ο κώδικας υπάρχει στο αρχείο *main.py*. Ο κώδικας περιέχει το κομμάτι ελιτισμού, ωστόσο, αν γίνουν σχόλιο οι τελευταίες 2 γραμμές στη συνάρτηση *mutate* ο κώδικας τρέχει κανονικά χωρίς ελιτισμό, όπως ζητείται πριν το ερώτημα B4.

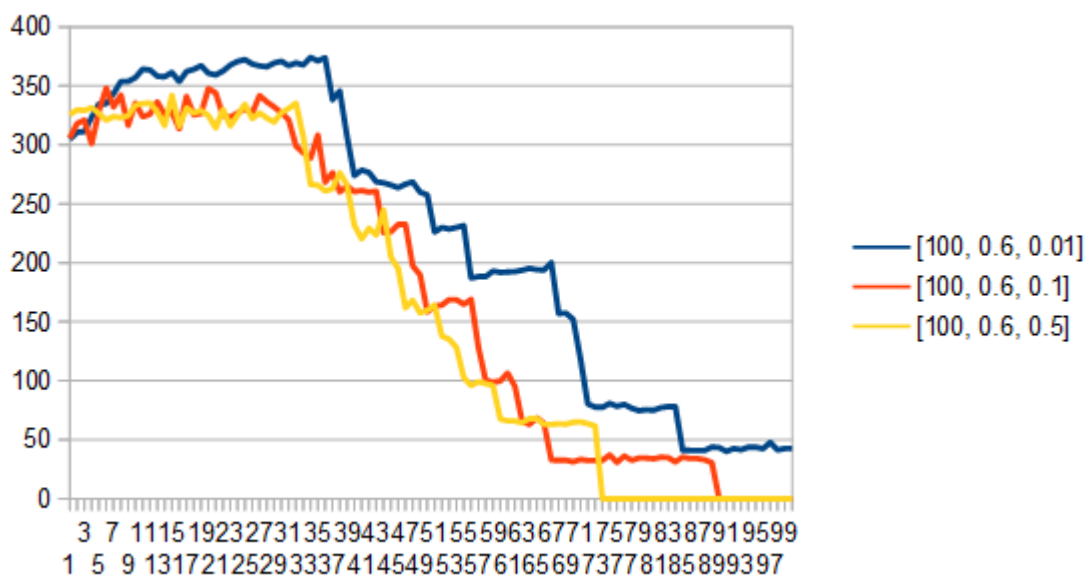
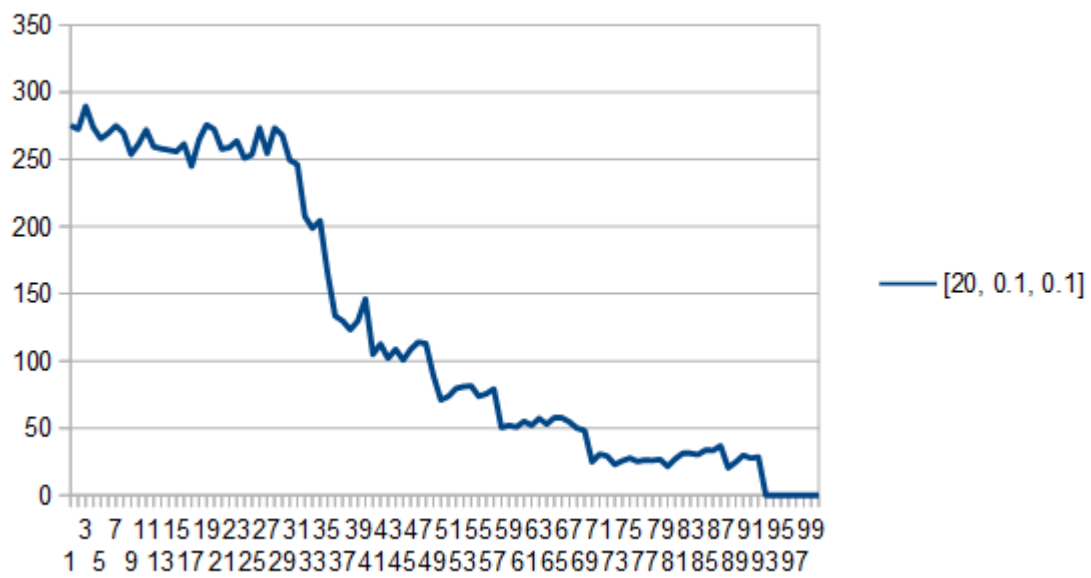
B3. Αξιολόγηση και Επίδραση Παραμέτρων

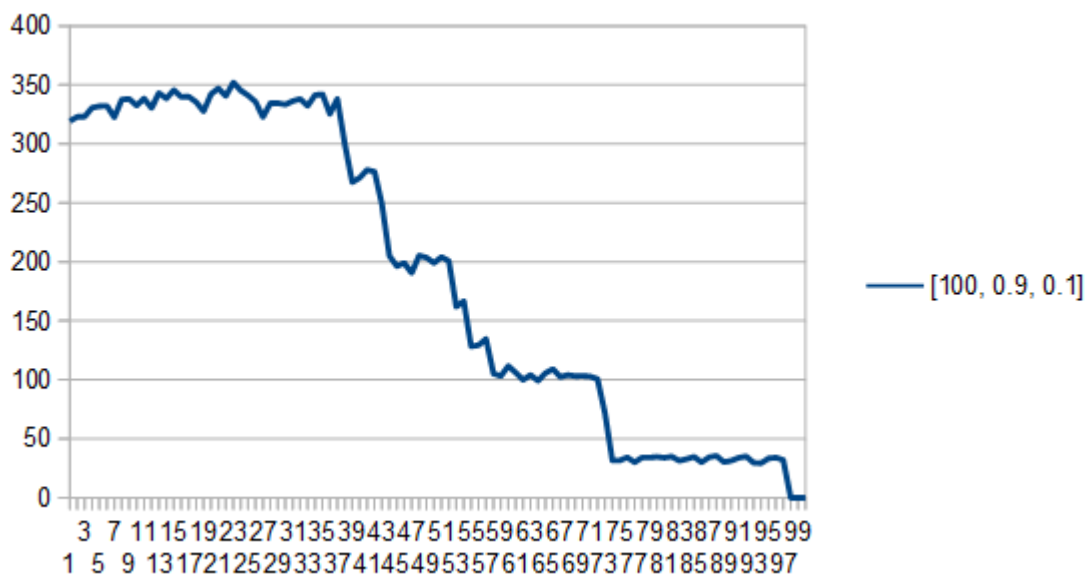
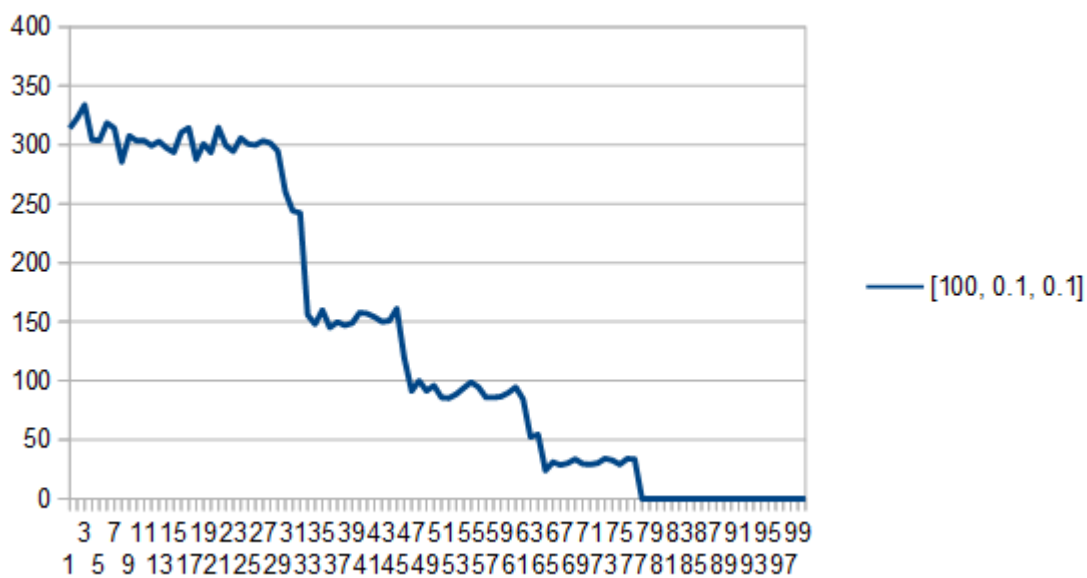
A. Πίνακας Παραμέτρων και Αποτελεσμάτων

A/A	ΜΕΓΕΘΟΣ ΠΛΗΘΥΣΜΟΥ	ΠΙΘΑΝΟΤΗΤΑ ΔΙΑΣΤΑΤΥΡΩΣΗΣ	ΠΙΘΑΝΟΤΗΤΑ ΜΕΤΑΛΛΑΞΗΣ	ΜΕΣΗ ΤΙΜΗ ΒΕΛΤΙΣΤΟΥ	ΜΕΣΟΣ ΑΡΙΘΜΟΣ ΓΕΝΕΩΝ
1	20	0.6	0.00	353.7	48.1
2	20	0.6	0.01	388.7	67.2
3	20	0.6	0.10	367.1	45.5
4	20	0.6	0.50	368.5	59.1
5	20	0.9	0.01	419.5	72.2
6	20	0.9	0.10	369.5	40.9
7	20	0.1	0.10	344.3	47.7
8	100	0.6	0.01	426.9	61.9
9	100	0.6	0.10	380.8	54.5
10	100	0.6	0.50	386.2	51.4
11	100	0.1	0.10	371.3	45.9
12	100	0.9	0.10	398.5	57.5

B. Καμπύλη Εξέλιξης







Γ. Συμπεράσματα

Γραφήματα. Τα γραφήματα φαίνεται να 'πέφτουν' μετά την εύρεση της καλύτερης λύσης. Αυτό συμβαίνει, επειδή οι καμπύλες σχηματίζονται χρησιμοποιώντας το μέσο όρο 10 εκτελέσεων, καθώς σε κάποιες εκτελέσεις προκύπτει *early stop*. Μέσα σε αυτόν το μέσο όρο προστίθενται μηδενικά, αφού το πρόγραμμα σταμάτησε νωρίτερα, κατεβάζοντας έτσι το μέσο όρο που προκύπτει στο γράφημα, δίνοντας την εντύπωση ότι η απόδοση πέφτει όσο μεγαλώνουν οι γενιές.

Συνθήκες Τερματισμού. Ως συνθήκες τερματισμού υλοποιήθηκαν οι 1, 3. Δεν υλοποιήθηκε η δεύτερη, διότι αν υλοποιούνταν, δεν θα σταματούσε ποτέ ο αλγόριθμος λόγω του πρώτου περιορισμού, αφού ο πρώτος χρειάζεται να μην βελτιωθεί ο καλύτερος για μερικές γενιές, ενώ ο δεύτερος το σταματάει στη πρώτη, η οποία δεν τον βελτιώνει αρκετά. Η υλοποίηση της δεύτερης συνθήκης τερματισμού δεν είναι κάτι το ιδιαίτερο. Ωστόσο, λόγω του αριθμού των μέγιστων γενεών (100) και της ορισμένης τιμής για *early stop* (30) στα περισσότερα αποτελέσματα ο αλγόριθμος τερματίζει στις 100 γενιές.

Μέγεθος Πληθυσμού. Μεγαλύτερο μέγεθος πληθυσμού φαίνεται να δίνει καλύτερο fitness. Επίσης, η καλύτερη λύση όλων εντοπίζεται στην εκτέλεση του αλγορίθμου με πλήθος 100, όπως είναι προφανές.

Πιθανότητα Διασταύρωσης. Μικρότερες πιθανότητες διασταύρωσης διαπιστώνεται ότι έχουν με μεγάλη διαφορά χειρότερα αποτελέσματα στην απόδοση του πληθυσμού.

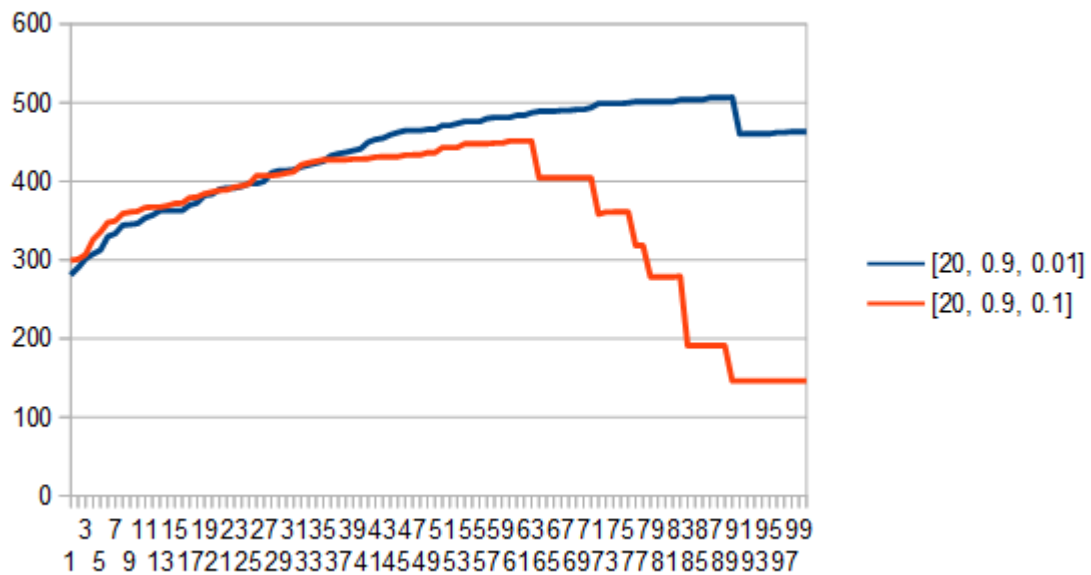
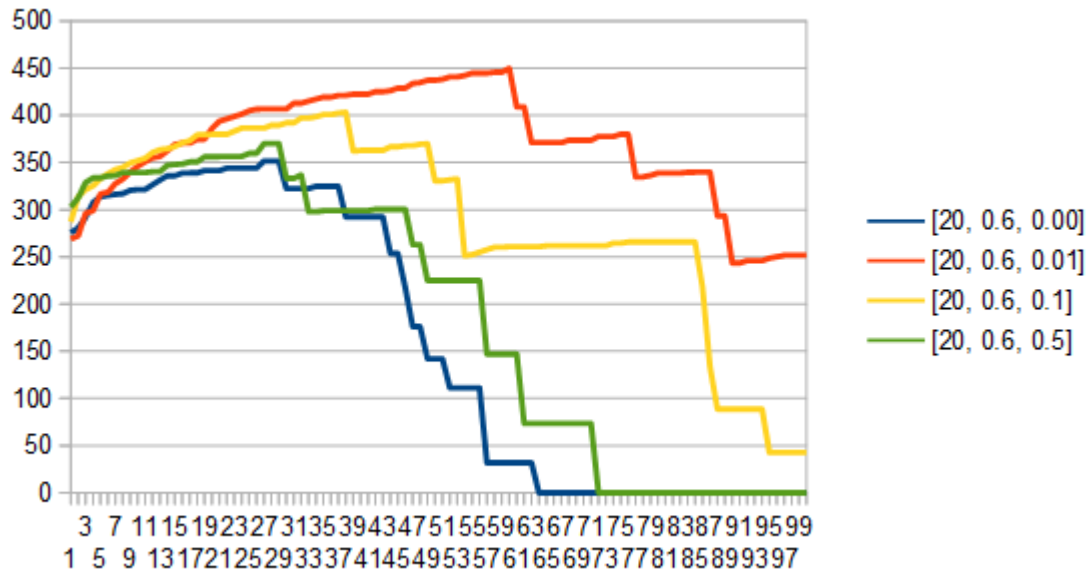
Πιθανότητα Μετάλλαξης. Από τον πίνακα φαίνεται ότι όσο μεγαλύτερη είναι η πιθανότητα μετάλλαξης τόσο χειρότερες είναι οι λύσεις, ενώ με μικρότερες τιμές τα αποτελέσματα είναι καλύτερα.

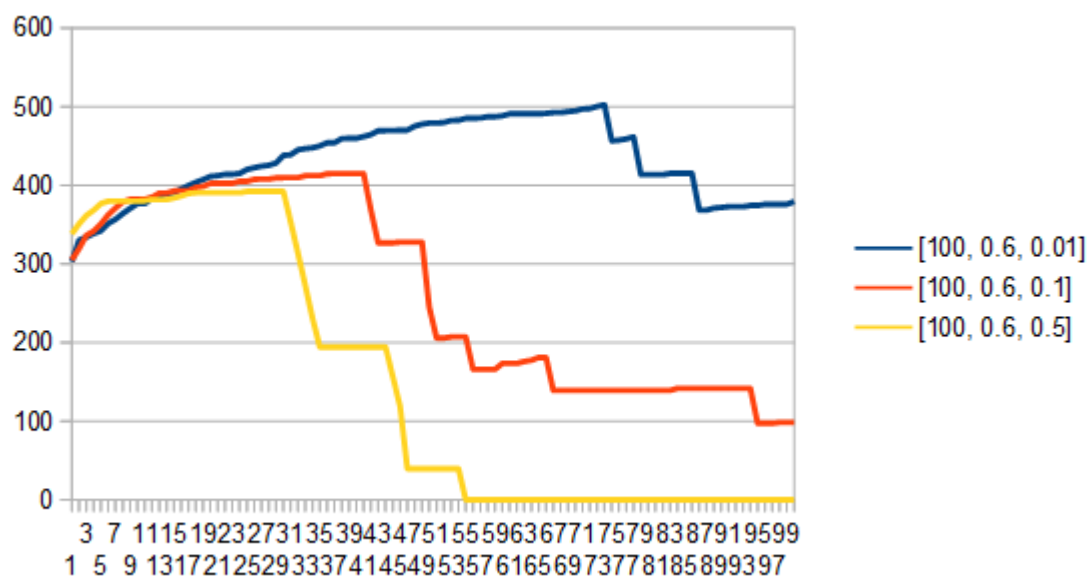
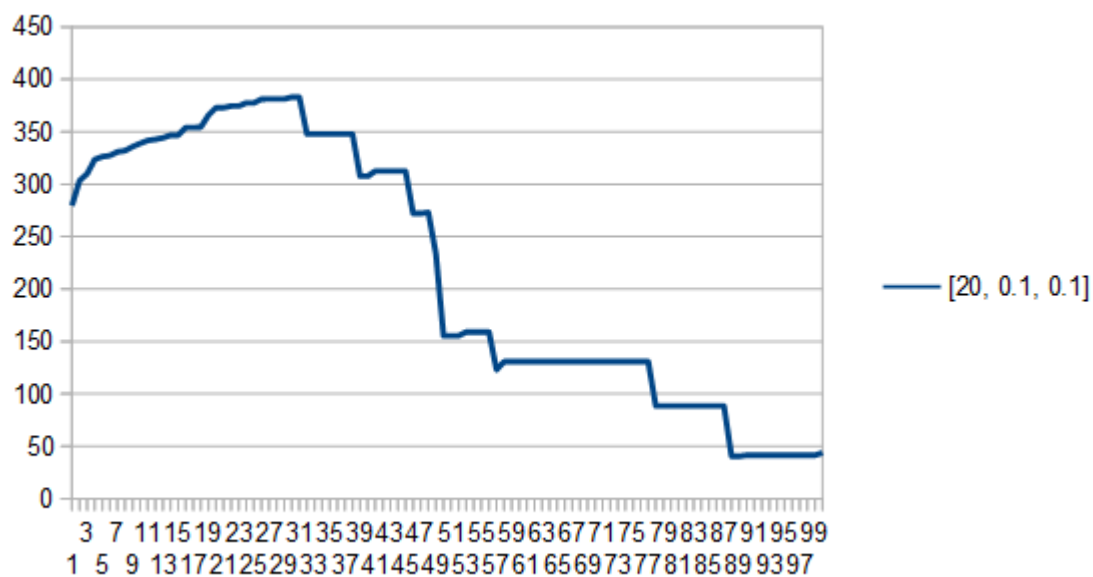
B4. Ελιτισμός

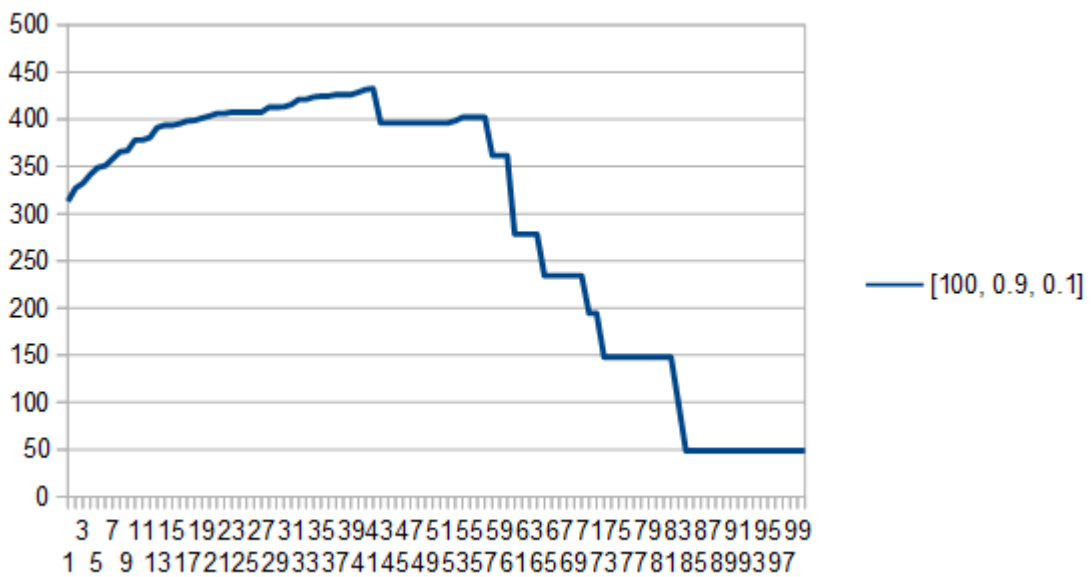
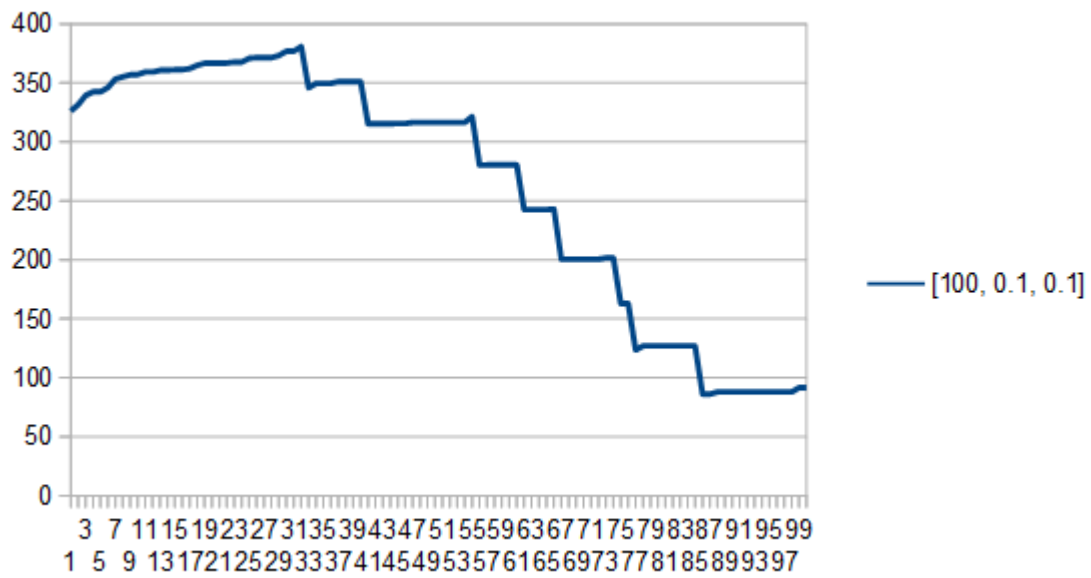
A. Πίνακας Παραμέτρων και Αποτελεσμάτων

A/A	ΜΕΓΕΘΟΣ ΠΛΗΘΥΣΜΟΥ	ΠΙΘΑΝΟΤΗΤΑ ΔΙΑΣΤΑΤΥΡΩΣΗΣ	ΠΙΘΑΝΟΤΗΤΑ ΜΕΤΑΛΛΑΞΗΣ	ΜΕΣΗ ΤΙΜΗ ΒΕΛΤΙΣΤΟΥ	ΜΕΣΟΣ ΑΡΙΘΜΟΣ ΓΕΝΕΩΝ
1	20	0.6	0.00	354.1	48.4
2	20	0.6	0.01	477.4	87.9
3	20	0.6	0.10	427.5	74.0
4	20	0.6	0.50	376.5	54.1
5	20	0.9	0.01	510.0	99.1
6	20	0.9	0.10	455.0	85.0
7	20	0.1	0.10	403.5	58.9
8	100	0.6	0.01	519.9	94.1
9	100	0.6	0.10	436.5	65.4
10	100	0.6	0.50	392.1	40.5
11	100	0.1	0.10	408.6	69.7
12	100	0.9	0.10	448.4	69.9

Β. Καμπύλη Εξέλιξης







Γ. Συμπεράσματα

Ο Ελιτισμός φαίνεται να βελτιώνει κατά πολύ όλες τις λύσεις, αλλά αυξάνει το πλήθος γενεών για την εύρεση τους.

Γραφήματα. Οι καμπύλες είναι πολύ πιο ομαλές και αύξουσες μέχρι το σημείο που παρατηρείται το φαινόμενο που περιγράφηκε στην προηγούμενη ενότητα, ενώ χωρίς ελιτισμό η απόδοση μπορεί να μειωθεί από γενιά σε γενιά.

Συνθήκες Τερματισμού. Στην περίπτωση του ελιτισμού η συνθήκη τερματισμού 2 ίσως ήταν πιο αποδοτική.

Δ. Υλοποίηση

Η υλοποίηση έγινε αρκετά απλά, εφόσον η ενημέρωση του καλύτερου του πληθυσμού γίνεται στην *evaluate_fitness* και η τελευταία διαδικασία που εκτελείται πριν από αυτήν είναι η *mutate*. Στο τέλος της *mutate* βγάζουμε το τελευταίο άτομο του πληθυσμού και προσθέτουμε τον *fitest*.