



<b>ΜΑΘΗΜΑ:</b>	<b>Βασικές Αρχές Προγραμματισμού</b>
<b>ΕΡΓΑΣΙΑ:</b>	<b>1<sup>η</sup></b>
<b>ΒΑΘΜΟΛΟΓΙΑ:</b>	<b>0 - 20</b>
<b>ΟΜΑΔΙΚΗ</b>	<b>ναι μέχρι 2 άτομα</b>
<b>ΗΜΕΡΟΜΗΝΙΑ ΑΝΑΚΟΙΝΩΣΗΣ:</b>	<b>17/12/2018</b>
<b>ΗΜΕΡΟΜΗΝΙΑ ΠΑΡΑΔΟΣΗΣ:</b>	<b>20/1/2019</b>
<b>ΤΡΟΠΟΣ ΠΑΡΑΔΟΣΗΣ:</b>	<b>Μέσω του elearning.auth.gr</b>

## Παιχνίδι Τρίλιζα (Tic Tac Toe)

Καλείστε να υλοποιήσετε στη γλώσσα C ένα πρόγραμμα με το οποίο θα μπορεί ο χρήστης να παίζει τρίλιζα απέναντι στον Η/Υ.

### Απαιτούμενα skills:

- Διαχείριση Αρχείων Κειμένου
- Δυναμική Διαχείριση Μνήμης
- Υπολογιστική Σκέψη (δομημένος προγραμματισμός με C)

### Περιγραφή

Να γίνει πρόγραμμα το οποίο θα δίνει τη δυνατότητα στο χρήστη να παίζει τρίλιζα απέναντι στον Η/Υ. Το πρόγραμμα θα πρέπει να περιλαμβάνει τα παρακάτω στοιχεία:

Μενού Επιλογών από το οποίο ο χρήστης θα επιλέγει αν α) θα παίζει νέο παιχνίδι απέναντι στον Η/Υ, β) θα παίζει νέο παιχνίδι δύο ατόμων (οι παίκτες παίζουν εναλλάξ), γ) θα δει τα στατιστικά για κάποιον παίκτη ή δ) θα τερματίσει το πρόγραμμα.

Κώδικα που θα παίζει τρίλιζα για τον Η/Υ. Θα πρέπει να φτιάξετε έναν αλγόριθμο που θα παίζει όσο το δυνατόν καλύτερα τρίλιζα (το ιδανικό είναι να μην χάνει ποτέ). Μπορείτε να χρησιμοποιήσετε εμπειρικούς κανόνες, ευριστικές συναρτήσεις, ή προαιρετικά να δοκιμάσετε αλγορίθμους σε παίγνια δύο ατόμων (MiniMax, Alpha Beta pruning)

Ένα εύχρηστο interface το οποίο θα παρουσιάζει το ταμπλό μετά από κάθε κίνηση, θα δίνει τη δυνατότητα επιλογής θέσης στην οποία θα βάλει ο χρήστης το επόμενο 'X' ή 'O' και θα ενημερώνει το χρήστη για την έκβαση του παιχνιδιού.

Δυνατότητα αποθήκευσης των στατιστικών για κάθε παίκτη. Το πρόγραμμα θα πρέπει να ζητάει το όνομα του παίκτη και να αποθηκεύει σε αρχεία κειμένου τα στατιστικά του (νίκες, ήττες, ισοπαλίες) και θα τα ενημερώνει μετά από κάθε παρτίδα.

### Δυναμική Διαχείριση Μνήμης

Τα δεδομένα που αφορούν στα στατιστικά των παικτών θα αποθηκεύονται προσωρινά στη μνήμη σε χώρο που θα δεσμεύεται δυναμικά και θα είναι ακριβώς ίσος με τον απαιτούμενο χώρο.

## Παρατηρήσεις

- Θα αξιολογηθούν: α) η ποιότητα του σχεδιασμού της λύσης σας, δηλαδή ο δομημένος – αρθρωτός σχεδιασμός του προγράμματος με χρήση συναρτήσεων, β) η εξοικονόμηση μνήμης χρησιμοποιώντας δυναμική δέσμευση, γ) η σταθερότητα του προγράμματος, δ) η έκθεση περιγραφής του προγράμματος και ε) ο καθαρός, ευανάγνωστος, δομημένος κώδικας
- Ο πηγαίος κώδικας του προγράμματος, το εκτελέσιμο αρχείο μαζί με ένα αρχείο pdf που θα περιγράφει την υλοποίησή σας να αποσταλούν μέσω του συστήματος [elearning.auth.gr](http://elearning.auth.gr)
- Ο κώδικας θα πρέπει να είναι επαρκώς σχολιασμένος.
- Τα προγράμματα θα υποβληθούν σε έλεγχο αντιγραφής (plagiarism detection) μέσω ειδικού προγράμματος για Code Similarity.
- Ορισμένοι ή όλοι οι φοιτητές που θα στείλουν εργασίες θα κληθούν σε προφορική εξέταση πάνω στην εργασία.
- Σε περίπτωση που φοιτητής στείλει εργασία και δεν δώσει πειστικές απαντήσεις στην προφορική εξέτασή της, ή διαπιστωθούν ομοιότητες της εργασίας του με άλλες εργασίες, ο βαθμός της εργασίας θα μηδενιστεί και ο φοιτητής θα παραπεμφθεί στη Γενική Συνέλευση του Τμήματος που θα αποφασίσει για περαιτέρω ποινές.