

# Αντικειμενοστρεφής Προγραμματισμός

Εργασία Ακαδημαϊκού Έτους 2016-2017

---

## Εισαγωγή

Στο πλαίσιο της εργασίας καλείστε να σχεδιάσετε και να υλοποιήσετε σε Java ένα παιχνίδι ερωτήσεων, βασισμένο σε ένα βραβευμένο παιχνίδι στο PlayStation, το [Buzz! Quiz World](#). Στόχος της εργασίας είναι να εξοικειωθείτε με τις διαδικασίες σχεδιασμού, υλοποίησης και ελέγχου αντικειμενοστρεφούς λογισμικού χρησιμοποιώντας τεχνικές απόκρυψης πληροφορίας, πολυμορφισμού και επαναχρησιμοποίησης κώδικα.

## Προδιαγραφές του Λογισμικού

Το παιχνίδι εκτυλίσσεται σε έναν αριθμό (π.χ. 6) γύρων (όχι σε πίτα), και ο κάθε γύρος περιλαμβάνει έναν αριθμό (π.χ. 5) ερωτήσεων πολλαπλής επιλογής με 4 απαντήσεις εκ των οποίων μόνο η μία είναι σωστή. Υπάρχουν δύο είδη ερωτήσεων: α) ερωτήσεις που συνοδεύονται και από εικόνα (π.χ. ποιος είναι ο εικονιζόμενος ηθοποιός;), και β) ερωτήσεις που δεν συνοδεύονται από εικόνα. Επίσης θα πρέπει να υπάρχει ένας αριθμός (π.χ. 4) από κατηγορίες ερωτήσεων (π.χ. αθλητικά, ιστορία, κτλ). Το παιχνίδι μπορεί να παίζεται είτε από έναν παίκτη, είτε ανταγωνιστικά από 2 παίκτες. Υπάρχουν διάφοροι τύποι γύρων:

- *Σωστή απάντηση.* Κάθε παίκτης που απαντάει σωστά κερδίζει 1000 πόντους.
- *Σταμάτησε το χρονόμετρο.* Υπάρχει ένα χρονόμετρο που μετράει αντίστροφα 5 δευτερόλεπτα και ο κάθε παίκτης που απαντάει σωστά κερδίζει τόσους πόντους όσους τα millisecond που απέμεναν ακόμα όταν απάντησε επί 0,2.
- *Ποντάρισμα.* Αρχικά εμφανίζεται η γενική κατηγορία. Ο παίκτης μπορεί να ποντάρει 250, 500, 750 και 1000 πόντους. Στη συνέχεια εμφανίζεται η ερώτηση και αν απαντήσει σωστά κερδίζει τους πόντους που πόνταρε, αλλιώς τους χάνει.
- *Γρήγορη απάντηση.* Ο πρώτος που απαντάει σωστά κερδίζει 1000 πόντους και ο δεύτερος 500 πόντους (αυτός ο τύπος ερώτησης έχει νόημα μόνο όταν παίζουν 2 παίκτες).
- *Θερμόμετρο.* Ο πρώτος παίκτης που θα απαντήσει σωστά 5 ερωτήσεις κερδίζει 5000 πόντους (αυτός ο τύπος ερώτησης έχει νόημα μόνο όταν παίζουν 2 παίκτες).

Το λογισμικό σας θα πρέπει να κρατάει αρχείο με τις νίκες κάθε παίκτη στο παιχνίδι 2 παιχτών, καθώς και με το υψηλότερο του σκορ στο ατομικό παιχνίδι και να παρουσιάζει αυτά τα στατιστικά όποτε ζητηθούν. Θα πρέπει επίσης να περιλαμβάνει τυχαιότητα (τυχαία σειρά εμφάνισης ερωτήσεων, αλλά και απαντήσεων ανά ερώτηση), καθώς και αποφυγή εμφάνισης της ίδιας ερώτησης στο ίδιο παιχνίδι. Οι ερωτήσεις καθώς και τα σχετικά δεδομένα τους (απαντήσεις, σωστή απάντηση, κατηγορία ερώτησης) θα πρέπει να είναι αποθηκευμένες σε αρχεία κειμένου.

Το λογισμικό σας θα πρέπει να είναι σχεδιασμένο βάσει των αρχών του αντικειμενοστρεφούς προγραμματισμού που παρουσιάζονται στο μάθημα. Θα πρέπει να αναλύσετε το λογισμικό σας σε έναν ικανό αριθμό κλάσεων, και όχι να βάλετε όλο τον κώδικα σε μια κλάση. Θα πρέπει να είναι σχεδιασμένο έτσι ώστε να είναι εύκολη η συντήρη και η επέκτασή του. Για παράδειγμα στο μέλλον μπορεί να θέλουμε να έχουμε και ερωτήσεις με ήχο, καθώς και ερωτήσεις με βίντεο. Μπορεί να θέλουμε να προστεθούν και άλλες κατηγορίες ερωτήσεων. Μπορεί επίσης να θέλουμε να προστεθεί η δυνατότητα να παίζουν ως 4 παίκτες. Οφείλετε να διαχωρίσετε την λογική του παιχνιδιού από τη διασύνδεση του με τον χρήστη (user interface), έτσι ώστε αν κάποιος θελήσει να αλλάξει τη διασύνδεση, αυτό να μην απαιτεί αλλαγές στη λογική. Το λογισμικό που θα αναπτύξετε θα πρέπει να έχει graphical user interface.

Για κάθε κλάση που θα αναπτύξετε, εκτός από τις κλάσεις που αφορούν τη διασύνδεση με τον χρήστη, θα πρέπει να υλοποιήσετε και αντίστοιχη κλάση ελέγχου. Αυτό μπορεί να γίνει είτε αναπτύσσοντας δικές σας κλάσεις ελέγχου είτε χρησιμοποιώντας το πλαίσιο ελέγχου JUnit.

Το λογισμικό θα πρέπει να είναι διεθνοποιημένο, ώστε να μπορεί να παίξει κανείς τόσο στα Ελληνικά όσο και στα Αγγλικά, ανάλογα με τις ρυθμίσεις συστήματος.

## Λογιστικά

Η εργασία χωρίζεται σε δύο επαναληπτικά μέρη ανάπτυξης της εργασίας, διάρκειας ενός μήνα το κάθε ένα, σε αρμονία με τις αρχές της ευέλικτης ανάπτυξης λογισμικού (agile software development) που έχει υιοθετήσει η βιομηχανία ανάπτυξης λογισμικού στις μέρες μας.

Στο πρώτο μέρος θα πρέπει να παραδώσετε μια πρώτη ελάχιστη έκδοση του λογισμικού σας, η οποία θα έχει command line interface, θα αφορά μόνο ατομικό παιχνίδι, θα υπάρχουν ερωτήσεις χωρίς εικόνα, θα υπάρχουν μόνο δύο τύποι γύρων (σωστή απάντηση, ποντάρισμα), θα είναι μόνο σε μία γλώσσα (Ελληνικά), δεν θα διατηρεί αρχείο με νίκες και σκορ. Ένας περιορισμένος αριθμός από ερωτήσεις (καθώς και τα σχετικά δεδομένα τους) θα είναι αποθηκευμένες στη μνήμη (στον κώδικα του προγράμματος). Το παραδοτέο αυτό θα μετρήσει κατά 10% στον τελικό βαθμό του μαθήματος.

Στο δεύτερο μέρος θα πρέπει να παραδώσετε την πλήρη έκδοση του λογισμικού σας. Το παραδοτέο αυτό θα μετρήσει κατά 20% στον τελικό βαθμό του μαθήματος.

Η εργασία θα πρέπει να γίνει από ομάδες των δύο ατόμων. Όσοι δεν μπορείτε να βρείτε δεύτερο άτομο, θα το δηλώσετε σε forum που υπάρχει στο e-learning για αυτόν τον σκοπό, έτσι ώστε να έρθετε σε επαφή μεταξύ σας και να δημιουργήσετε ομάδα. Η καταληκτική ημερομηνία αποστολής κάθε παραδοτέου φαίνεται στο χρονοδιάγραμμα του μαθήματος. Εκπρόθεσμη υποβολή μέχρι 2 ημέρες θα έχει ως αποτέλεσμα τη μείωση του βαθμού κατά 10% την ημέρα. Θα πρέπει να σταλεί μέσω του [elarning.auth.gr](mailto:elarning.auth.gr), ένα αρχείο zip με όνομα **AEM1-AEM2.zip**, το οποίο θα περιέχει:

1. Τον πηγαίο κώδικα που αναπτύξατε για την εφαρμογή καθώς και για τον έλεγχό της. Ο κώδικας θα πρέπει να είναι τεκμηριωμένος με σχόλια τύπου Javadoc.
2. Ένα jar αρχείο με το bytecode της εφαρμογής σας.

Προσοχή: στο σύστημα [elarning.auth.gr](mailto:elarning.auth.gr) μπορεί να ανεβεί **ένα** μόνο αρχείο ανά χρήστη. Αν ανεβεί δεύτερο, **αντικαθίσταται το πρώτο**. Έτσι, αν ξεχάσατε κάποιο αρχείο και θέλετε να το συμπληρώσετε, θα πρέπει να ξαναστείλετε το .zip αρχείο σας – όχι μόνο αυτό που έλειπε. Οι εργασίες θα πρέπει να υποβάλλονται από έναν εκπρόσωπο της ομάδας και μόνο.

## Αξιολόγηση

Για την αξιολόγηση των εργασιών θα συνεκτιμηθεί:

1. Η σχεδίαση του λογισμικού (ευκολία συντήρησης και επέκτασης)
2. Η υλοποίηση του λογισμικού (πληρότητα υλοποίησης λειτουργιών, ποιότητα υλοποίησης)
3. Ο έλεγχος του λογισμικού (πληρότητα και ποιότητα κλάσεων ελέγχου)
4. Η τεκμηρίωση εντός του κώδικα
5. Η επίδοση κάθε φοιτητή/φοιτήτριας στην προφορική εξέταση της εργασίας

Προϊόντα αντιγραφής θα μηδενίζονται και όλοι οι εμπλεκόμενοι φοιτητές θα παραπέμπονται στη γενική συνέλευση του Τμήματος (βλέπε σελίδα 22 του οδηγού σπουδών).