

Εργασία στο μάθημα Αντικειμενοστραφής Προγραμματισμός

Τμήμα Πληροφορικής και Τηλεπικοινωνιών (Άρτα)

Πανεπιστήμιο Ιωαννίνων

Χρήστος Γκόγκος

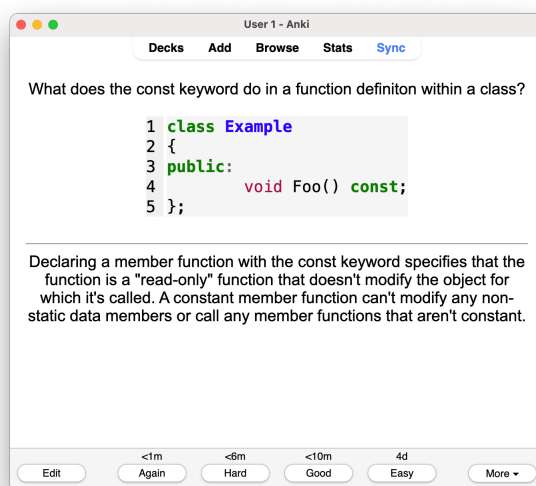
11 Νοεμβρίου 2024

1 Εισαγωγή

Σκοπός της εργασίας είναι η εξοικείωση με τον αντικειμενοστραφή προγραμματισμό και με τον τρόπο που η γλώσσα προγραμματισμού C++ χρησιμοποιείται στα πλαίσια του αντικειμενοστραφούς προγραμματισμού. Αυτό επιδιώκεται να συμβεί μέσω της κατασκευής μιας ολοκληρωμένης εφαρμογής.

2 Εκφώνηση εργασίας

Ζητούμενο της εργασίας είναι η κατασκευή μιας εφαρμογής, παρόμοιας με τη δημοφιλή εφαρμογή Anki¹, που το όνομά της στα Ιαπωνικά σημαίνει απομνημόνευση. Η εφαρμογή Anki, όπως φαίνεται στο Σχήμα 1, χρησιμοποιεί κάρτες μνήμης (flashcards) και στηρίζεται σε τεχνικές που η γνωσιακή επιστήμη (cognitive science) θεωρεί ότι λειτουργούν καλά στην απόκτηση και στη διατήρηση γνώσης. Τέτοιες τεχνικές είναι η χρονική επανάληψη (spaced repetition [1]) και η ενεργή ανάκληση (active recall²). Εξοικειωθείτε με τη λειτουργία του Anki κατεβάζοντας την εφαρμογή και πειραματιστείτε με τις δυνατότητές του.



Σχήμα 1: Ένα παράδειγμα κάρτας μνήμης (flashcard) του Anki.

¹<https://apps.ankiweb.net/index.html>

²<https://subjectguides.york.ac.uk/study-revision/active-recall>

Περιγραφή της εφαρμογής Ζητείται να κατασκευάσετε μια εφαρμογή που να επιτρέπει στους χρήστες:

- Να δημιουργούν, αποθηκεύουν και οργανώνουν συλλογές από κάρτες μνήμης.
- Να μελετούν τις κάρτες χρησιμοποιώντας την τεχνική της χρονικής επανάληψης.
- Να παρακολουθούν την πρόοδό τους.

Λειτουργίες εργασίας Η εφαρμογή πρέπει να υλοποιηθεί με αντικειμενοστραφή προσέγγιση και να περιλαμβάνει τις παρακάτω λειτουργίες:

- Δημιουργία καρτών: Κάθε κάρτα να έχει μία ερώτηση και μία απάντηση και οι χρήστες να μπορούν να δημιουργούν, να επεξεργάζονται και να διαγράφουν κάρτες.
- Δημιουργία συλλογών καρτών (decks): Οι κάρτες να μπορούν να οργανωθούν σε θεματικές συλλογές και κάθε συλλογή να μπορεί να περιέχει πολλαπλές κάρτες, ενώ κάθε κάρτα να μπορεί να ανήκει σε μία ή περισσότερες συλλογές.
- Μελέτη με χρονική επανάληψη: Κατά τη μελέτη, ο χρήστης να βλέπει μία κάρτα και απαντά αν την γνωρίζει ή όχι. Ανάλογα με την απάντηση, το πρόγραμμα θα προγραμματίζει την επανεμφάνιση της κάρτας (σε μικρότερο ή μεγαλύτερο χρονικό διάστημα). Η στρατηγική επανάληψης μπορεί να είναι απλή (π.χ., να αυξάνει το χρόνο επανάληψης ανάλογα με τον αριθμό των σωστών απαντήσεων).
- Παρακολούθηση προόδου: Το σύστημα να παρακολουθεί και να αποθηκεύει την πρόοδο του χρήστη σε κάθε κάρτα.

Τεχνικές απαιτήσεις Οι τεχνικές απαιτήσεις της εφαρμογής είναι οι ακόλουθες:

- Αντικειμενοστραφής σχεδίαση: Ορίστε κατάλληλες κλάσεις και αντικείμενα για τη διαχείριση των καρτών, των συλλογών και της προόδου του χρήστη.
- Αποθήκευση δεδομένων: Υλοποιήστε έναν απλό μηχανισμό αποθήκευσης δεδομένων, είτε σε αρχεία (π.χ., JSON, CSV) είτε σε μια βάση δεδομένων SQLite. Τα δεδομένα πρέπει να αποθηκεύονται και να επαναφορτώνονται κατά την εκκίνηση της εφαρμογής, ώστε να μην χάνονται όταν ο χρήστης κλείνει το πρόγραμμα.
- Μοναδιαίοι έλεγχοι (unit tests): Προσθέστε μοναδιαίους ελέγχους που να ελέγχουν όλη τη λειτουργικότητα της εφαρμογής.
- Διεπαφή χρήστη: Δημιουργήστε διεπαφή έτσι ώστε η εφαρμογή να μπορεί να χρησιμοποιηθεί εύκολα και διαισθητικά. Η διεπαφή μπορεί να είναι είτε γραφική (GUI) είτε γραμμής εντολών (CLI). Αν επιλέξετε GUI, μπορείτε να χρησιμοποιήσετε βιβλιοθήκες όπως η wxWidgets³.
- Διάγραμμα κλάσεων: Σχεδιάστε ένα διάγραμμα κλάσεων UML που να απεικονίζει όλες τις κλάσεις της εφαρμογής και τις συσχετίσεις μεταξύ τους.

2.1 Προαιρετικές λειτουργίες - επιπλέον βαθμολόγηση μέχρι +20%

Οι φοιτητές και οι φοιτήτριες που αναζητούν προγραμματιστικές προκλήσεις μπορούν να υλοποιήσουν επιπλέον χαρακτηριστικά όπως τα ακόλουθα:

- Υποστήριξη πολλών χρηστών με δυνατότητα να φορτώνουν και να αποθηκεύουν το προφίλ τους.
- Προσθήκη οπτικών και ηχητικών εφέ.
- Ιδιωτικό αποθετήριο στο github⁴ με τον κώδικα της εργασίας (διαμοιρασμός με τον χρήστη csgogos@uoi.gr). Το αποθετήριο μπορεί να γίνει δημόσιο μετά την ολοκλήρωση βαθμολόγησης της εργασίας.
- Εκτέλεση της εφαρμογής σε Docker⁵ container.
- Δημιουργία μιας συλλογής τουλάχιστον 50 καρτών για διευκόλυνση της μελέτης του αντικειμενοστραφούς προγραμματισμού με τη C++, στα ελληνικά, και μεταφορά στην εφαρμογή της εργασίας, αλλά και στο Anki.

³<https://www.wxwidgets.org/>

⁴<https://github.com/>

⁵<https://www.docker.com/>

2.2 Κριτήρια βαθμολόγησης

Τα κριτήρια βαθμολόγησης της εργασίας είναι τα ακόλουθα:

- Σωστή εφαρμογή αρχών αντικειμενοστραφούς σχεδίασης και προγραμματισμού με τη C++. (30%)
- Λειτουργικότητα και αξιοπιστία. (30%)
- Σχεδίαση και υλοποίηση διεπαφής χρήστη. (20%)
- Έγερξη unit tests με το catch2⁶ ή με άλλο testing framework της C++, για τις επιμέρους λειτουργίες της εφαρμογής. (10%)
- Διάγραμμα κλάσεων UML. (10%)

3 Υποβολή εργασίας

Σχετικά με την υποβολή της εργασίας ισχύουν τα ακόλουθα:

- Η εργασία μπορεί να υποβληθεί μόνο στο ecourse: <https://ecourse.uoi.gr/course/view.php?id=1944>.
- Η προθεσμία υποβολής της εργασίας είναι στις 12/1/2025.
- Η γλώσσα υλοποίησης της εργασίας είναι η C++.
- Παραδοτέα εργασίας (σε ένα zip αρχείο):
 - Κώδικας με επαρκή σχολιασμό.
 - Οδηγίες μεταγλώττισης και εκτέλεσης σε ένα αρχείο README.md.
 - Έγγραφο με screenshots από την εκτέλεση της εφαρμογής.

Αναφορές

- [1] Sean HK Kang. “Spaced repetition promotes efficient and effective learning: Policy implications for instruction”. Στο: *Policy Insights from the Behavioral and Brain Sciences* 3.1 (2016), σσ. 12–19.

⁶<https://github.com/catchorg/Catch2>