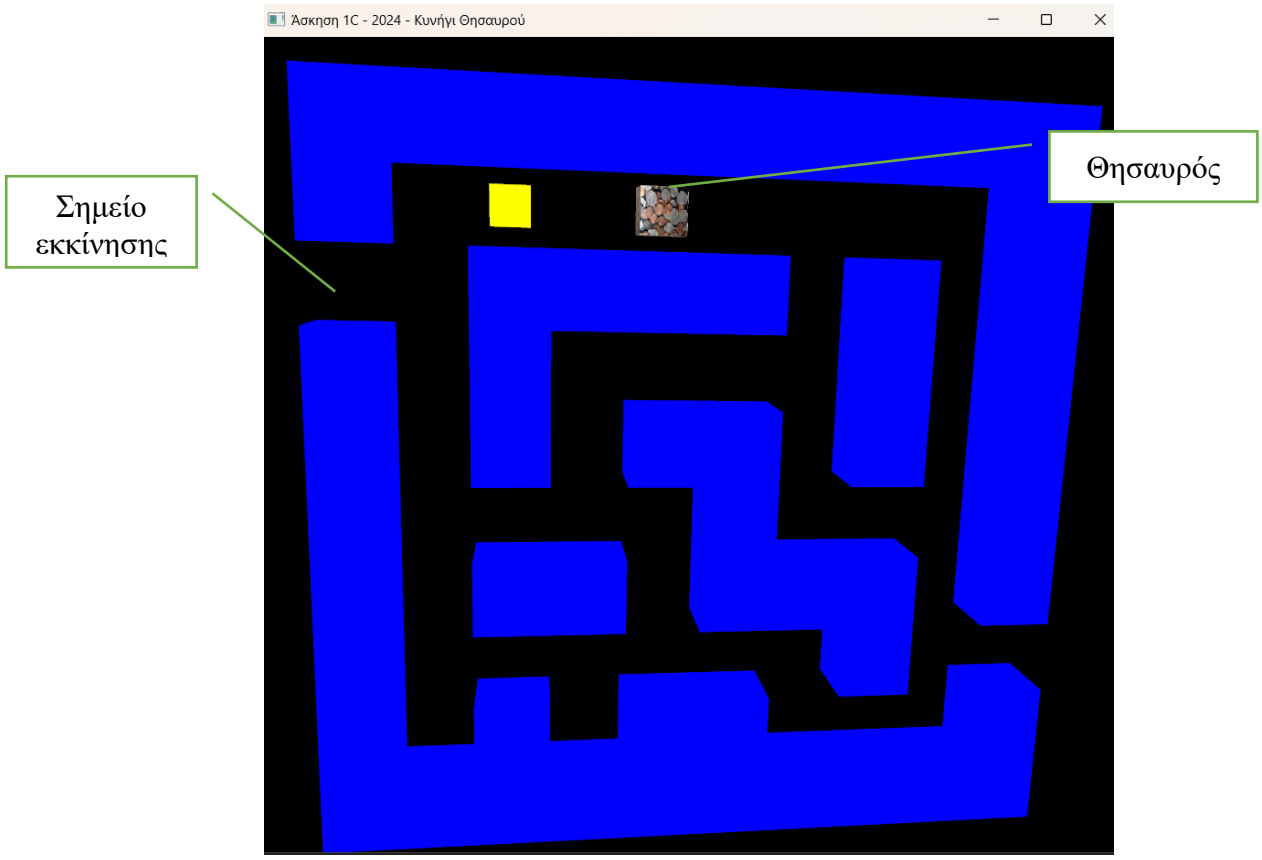
Πανεπιστήμιο Ιωαννίνων
Τμήμα Μηχ. Η/Υ και Πληροφορικής

Γραφικά Υπολογιστών και Συστήματα Αλληλεπίδρασης
Ακαδημαϊκό Έτος 2024-2025

ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΗ ΑΣΚΗΣΗ 1-Γ

Σκοπός του τρίτου μέρους του Συνόλου Προγραμματιστικών Ασκήσεων OpenGL είναι να εξασκηθείτε στη χρήση βασικών βιβλιοθηκών στοιχειωδών γραφικών της Opengl 3.3 (και μεταγενέστερων εκδόσεων) οι οποίες υποστηρίζουν 2Δ και 3Δ γραφικά. Στην άσκηση αυτή θα δημιουργήσετε μια εφαρμογή-παιχνίδι στο οποίο ένας χαρακτήρας θα κυνηγάει «θησαυρούς».



Εικόνα 1 – Τυχαίο στιγμιότυπο της εφαρμογής-παιχνιδιού

Η εργασία 1Γ είναι συνέχεια της άσκησης 1Β. Πιο συγκεκριμένα:

(i) (5%) Φτιάξτε ένα πρόγραμμα που θα ανοίγει ένα βασικό παράθυρο **950x950**. Το background του παραθύρου στην περιοχή εργασίας να είναι μαύρο. Το παράθυρο θα έχει τίτλο «Εργασία 1Γ – 2024 – Κυνήγι Θησαυρού» (με ελληνικούς χαρακτήρες – όχι greeklish). Με το πλήκτρο **SPACE** η εφαρμογή τερματίζει.

(ii) (5%) Ζωγραφίστε τον λαβύρινθο και τον κινούμενο χαρακτήρα Α, όπως περιγράφονται στα ερωτήματα (ii) και (iii) της εκφώνησης 1Β, δηλαδή:

- Ο λαβύρινθος σχηματίζεται ζωγραφίζοντας κύβους μπλε χρώματος, που αντιστοιχούν στα τοιχώματα του λαβύρινθου, οπότε μπορεί να αναπαρασταθεί από έναν διδιάστατο πίνακα που περιέχει τιμές 0 ή 1. Η τιμή "1" αντιστοιχεί σε τοίχος, ενώ η τιμή "0" αντιστοιχεί σε μονοπάτι. Το πρόγραμμά σας ζωγραφίζει τους τοίχους του λαβύρινθου είτε ζωγραφίζοντας έναν-έναν τους κύβους, είτε φτιάχνοντας παραλληλεπίπεδα (=ομαδοποίηση κύβων) χρησιμοποιώντας κατάλληλα τρίγωνα. Η κάτω πλευρά του λαβύρινθου «κάθεται» πάνω στο επίπεδο xy (δηλαδή όπου z=0) και το κέντρο της κάτω πλευράς είναι το σημείο (0,0,0) του επιπέδου. Κάθε κύβος έχει πλευρά μήκους 1

- Ο κινούμενος χαρακτήρας **A**, που διασχίζει τον λαβύρινθο, είναι ένας κύβος με μήκος πλευράς 0.5 και κίτρινο χρώμα. Το κέντρο του **A** συμπίπτει με το κέντρο του κύβου του λαβύρινθου στο οποίο είναι τοποθετημένο. Η αρχική θέση του **A** είναι στην είσοδο του λαβύρινθου.

(iii) (30%) Μέσα στον λαβύρινθο θα εμφανίζεται, περιοδικά και για ορισμένο χρονικό διάστημα, ένα αντικείμενο **B** (θησαυρός) σε τυχαία θέση κάθε φορά μέσα στο λαβύρινθο. Η θέση αυτή δεν μπορεί να συμπίπτει με θέση στην οποία βρίσκεται εκείνη τη στιγμή ο κινούμενος χαρακτήρας **A** και δεν μπορεί να είναι σε θέση όπου βρίσκεται τοίχος. Ο θησαυρός θα εμφανίζεται σταθερά για ένα χρονικό διάστημα και μετά θα εξαφανίζεται και θα εμφανίζεται κατευθείαν σε άλλη τυχαία θέση. Το κέντρο του **B** συμπίπτει με το κέντρο του κύβου του λαβύρινθου στο οποίο εμφανίζεται.

Ο θησαυρός **B** είναι ένας κύβος με μήκος πλευράς 0.8. Στο μοντέλο **B** θα εφαρμόσετε την υφή coins.jpg (σας δίνεται το αντίστοιχο αρχείο). Θα χρειαστεί να υπολογίσετε τις uv συντεταγμένες του B.

(iv) (30%) Ο χρήστης προσπαθεί να κινήσει τον κινούμενο χαρακτήρα **A** έτσι ώστε να προλάβει να «ακουμπήσει» τον θησαυρό, όπου αυτός βρίσκεται πριν αλλάξει θέση. Όταν ακουμπήσει ο κινούμενος χαρακτήρας τον θησαυρό, τότε αυτός συρρικνώνεται στο μισό του μέγεθος και μετά εξαφανίζεται. Η υλοποίησή σας να είναι τέτοια ώστε να προλαβαίνει ο χρήστης να δει την συρρίκνωση πριν ο θησαυρός εξαφανιστεί.

(v) (20%) Η λειτουργία κάμερας του προγράμματος θα είναι αυτή της Άσκησης **1B**, μαζί με μία συμπλήρωση. Πιο αναλυτικά: Να υλοποιήσετε μια κάμερα που θα ελέγχεται μόνο με τα πλήκτρα του πληκτρολογίου (*να γίνεται έλεγχος μόνο για key press*).
Η κάμερα θα κινείται στους άξονες του παγκόσμιου συστήματος συντεταγμένων με τους εξής τρόπους:

- γύρω από τον άξονα x με τα πλήκτρα <**w**> και <**x**>
- γύρω από τον άξονα y με τα πλήκτρα <**q**> και <**z**>
- κάνει zoom in/zoom out με κατεύθυνση το κέντρο του λαβύρινθου με τα πλήκτρα <+> και <-> του numerical keypad του πληκτρολογίου
- μετακινείται στον άξονα x (panning) με τα πλήκτρα **<g>** και **<h>**
- μετακινείται στον άξονα y (panning) με τα πλήκτρα **<t>** και **<b>**

Η κάμερά σας να είναι τοποθετημένη αρχικά στο σημείο (0.0, 0.0, 20.0) ώστε να κοιτάει προς το σημείο **E**(0,0,0.25) με ανιόν διάνυσμα (up vector) το (0.0, 1.0, 0.0).
(Σημείωση: αφού ορίσετε τιμή για το FOV (field of view), αυτή δεν θα πρέπει να αλλάζει κατά την διάρκεια εκτέλεσης του προγράμματος).

(vi) (10%) Θα ΠΡΕΠΕΙ ΑΠΑΡΑΙΤΗΤΑ ΝΑ ΥΠΑΡΧΕΙ ΕΝΑ ΑΡΧΕΙΟ ***"readme.pdf"*** που θα περιέχει τα ονοματεπώνυμα και ΑΜ των μελών της ομάδας, αναλυτικές πληροφορίες για την λειτουργία του προγράμματος και ιδιαίτερα για όποιες ιδιαιτερότητες, προβλήματα ειδικές συνθήκες, και άλλες πληροφορίες για τον κώδικα κτλ. Να χρησιμοποιήσετε το υπόδειγμα που σας δίνεται.

**BONUS

(α) (20 μονάδες) Στον θησαυρό B, κάθε φορά που θα εμφανίζεται, να εφαρμόζεται τυχαία μία από 3 διαφορετικές υφές.

(β) (30 μονάδες) Προσθέστε οπτικά και ηχητικά εφέ όταν ο κινούμενος χαρακτήρας A ακουμπήσει τον θησαυρό.

(γ) (25 μονάδες) Προσθέστε φωτισμό μοντέλου phong, υλοποιώντας το μοντέλο phong στον fragment shader με μία φωτεινή σημειακή πηγή στο (10.0, 8.0, 4.0).

(δ) (25 μονάδες) Προσθέστε έξι πλήκτρα για τη μετακίνηση της φωτεινής πηγής του ερωτήματος (β) με τον ίδιο τρόπο που γίνεται η μετακίνηση της κάμερας του ερωτήματος (v).

Παράδοση:

Η άσκηση θα παραδοθεί ηλεκτρονικά την **Δευτέρα, 02/12/2024** στις 9 μμ.

Κάντε ένα αντίγραφο του αρχείου κώδικά σας από την άσκηση 1Β και μετονομάστε το σε **Source-1C.cpp.** Εκεί θα υλοποιήσετε την άσκηση. Μπορείτε να χρησιμοποιήσετε μόνο τις βιβλιοθήκες γραφικών GLFW, GLEW και GLM.

Οδηγίες για την παράδοση υπάρχουν στην ηλεκτρονική σελίδα του ecourse του μαθήματος. Οι ασκήσεις ελέγχονται για κοινό κώδικα και αντιγραφή. Τέτοιες περιπτώσεις μηδενίζονται.

Η άσκηση εκπονείται και παραδίδεται σε ομάδες των δυο (το πολύ) ατόμων. Ο τρόπος βαθμολόγησης είναι αυστηρός και ίδιος είτε είστε σε ομάδα, είτε είστε μόνοι σας.

Το C αυτό μέρος του πρώτου συνόλου προγραμματιστικών ασκήσεων μετράει 15% στη βαθμολογία του μαθήματος. Υπενθυμίζουμε ότι στο μάθημα θα πρέπει να πάρετε τουλάχιστον 40/100 στο σύνολο της βαθμολογίας του πρώτου συνόλου των προγραμματιστικών ασκήσεων. Ο βαθμός του πρώτου συνόλου προγραμματιστικών ασκήσεων δίνεται από τον τύπο:

*(βαθμός πρώτου συνόλου προγραμματιστικών ασκήσεων) = (βαθμός Α μέρους) * 1/6 + (βαθμός Β μέρους) * 1/3 + (βαθμός Γ μέρους) * 1/2*