

ΓΡΑΦΙΚΑ  
REPORT 2ΗΣ ΕΡΓΑΣΙΑΣ ΣΕ UNITY

# 3D Maze

ΚΑΛΟΥΔΗΣ ΣΠΥΡΙΔΩΝ	2447
ΓΚΙΤΣΑΚΗΣ ΔΗΜΟΣ	2425

## ΕΙΣΑΓΩΓΗ

Το project μας αποτελείται από δύο scenes, τα Game και GameOver. Επίσης περιέχει 6 διαφορετικά scripts τα οποία υλοποιούν όλο το παιχνίδι και όλα τα ερωτήματα της άσκησης. Για να ξεκινήσει το παιχνίδι διαλέγουμε το scene “Game” και πατάμε το κουμπί run.

## SCENES

- **Game**

Το scene “Game” είναι βασισμένο στο scene των αρχείων του Φροντιστηρίου «4.Keys».

Περιέχει ένα αντικείμενο FPSController (το οποίο με τη σειρά του έχει attached το Pickaxe και μια πηγή φωτός που ακολουθεί τον παίκτη), μία εξωτερική πηγή φωτός (“Directional Light”), το αντικείμενο “Canvas” το οποίο έχει το κείμενο που βλέπει πάνω δεξιά ο παίκτης (σκορ και pickaxes), ένα αντικείμενο που αντιπροσωπεύει τα τυχαία σφυριά που πέφτουν στο σπάσιμο ενός block και την εξωτερική-εναλλακτική κάμερα (“Outside Camera”). Τέλος, έχει 3 αντικείμενα ήχου, τα “PickaxeHit”, “BlockBreak”, “Teleport”.

- **GameOver**

Το scene “GameOver” είναι βασισμένο στο scene των αρχείων του Φροντιστηρίου «1.Window».

Περιέχει την μία και βασική κάμερα “Main Camera”, μία εξωτερική πηγή φωτισμού «Directional Light” και το αντικείμενο “Canvas” που έχει attached το κείμενο που βλέπει ο χρήστης στην οθόνη λήξης του παιχνιδιού.

# SCRIPTS

## 1. Game

Με το που τρέξουμε το παιχνίδι (Scene “Game”), ενεργοποιείται η συνάρτηση **Start( )** του script “Game”. Στις γραμμές 40-63 υλοποιείται η προσπέλαση του αρχείου file.maz το οποίο βρίσκεται στον φάκελο του project. Έπειτα, αφού διαβαστούν τα απαραίτητα στοιχεία από το αρχείο και αρχικοποιηθούν οι κατάλληλες μεταβλητές, με τη χρήση 6 for στις γραμμές 67-131 δημιουργούνται αόρατα στο χρήστη κουτιά πάνω, κάτω, δεξιά και αριστερά των ακριανών κύβων του maze τα οποία λειτουργούν ως όρια για το μέχρι που μπορεί να κινηθεί ο παίκτης. Με μία τριπλή for και χρησιμοποιώντας το αρχείο που διαβάσαμε, στις γραμμές 133-206 δημιουργούνται οι  $n*n*1$  κύβοι κάνοντας χρήση του script Cubes και συγκεκριμένα της μεθόδου createCube(), οι οποίοι κύβοι παίρνουν το σωστό χρώμα/texture σύμφωνα με το file.maz. Μέσα στην τριπλή for κρατούνται επίσης οι κενές θέσεις (E) του πρώτου επιπέδου και στις γραμμές 207-210 με τη χρήση της Random διαλέγουμε ένα από αυτά τα κενά και τοποθετούμε τον παίκτη εκεί.

Σε κάθε frame καλείται η μέθοδος **Update( )** η οποία:

- Στις γραμμές 217-223 ελέγχει αν υπάρχουν διαθέσιμα pickaxes ώστε να το εμφανίσει/εξαφανίσει.
- Στις γραμμές 224-228 ενημερώνει την ώρα που πέρασε από την αρχή του παιχνιδιού και ανανεώνει το κείμενο που υπάρχει πάνω δεξιά ώστε να γνωρίζει ο χρήστης το σκορ και πόσα pickaxes απομένουν. Για τον υπολόγισμό του σκορ καλείται η βοηθητική συνάρτηση **calculateScore( )**.

- Στις γραμμές 230-244 αρχικοποιείται και υλοποιείται το raycasting το οποίο χρησιμοποιούμε για να βρούμε ποιον κύβο κοιτάει ο παίκτης ανά πάσα στιγμή.
- Στις γραμμές 248-301 υλοποιείται το «χτύπημα» του παίκτη. Ο παίκτης μπορεί να χρησιμοποιήσει το σφυρί του και να χτυπήσει έναν κύβο με το πλήκτρο H ή με το αριστερό κλικ του ποντικιού. Στην γραμμή 251 ενεργοποιείται η κίνηση του pickaxe στην οθόνη και στην γραμμή 254 αναγνωρίζεται το αντικείμενο το οποίο χτύπησε ο παίκτης με χρήση raycast. Αν χτύπησε κύβο τότε καλείται η συνάρτηση **PlayPickaxeHit( )** η οποία κάνει τον ήχο του χτυπήματος πάνω σε κύβο, το pickaxe χάνει 10 αντοχή και υπολογίζονται τα πόσα pickaxes έχουν χρησιμοποιηθεί (για το σωστό υπολογισμό σκορ). Έπειτα καλείται η βοηθητική συνάρτηση **colorPickaxe( )** η οποία αλλάζει το χρώμα του σφυριού ανάλογα με την πόση αντοχή του απομένει (Αν έχει 70+ αντοχή το σφυρί είναι κόκκινο, αν έχει 40+ αντοχή το σφυρί είναι μπορντό, ενώ κάτω από αυτό το σφυρί είναι μαύρο). Για να βαφτεί το σφυρί το αντίστοιχο χρώμα, στο φάκελο Assets/Resources υπάρχουν τα 3 κατάλληλα materials (Red, Bordo, Black). Έπειτα εφόσον χτυπήθηκε κύβος, του αφαιρεί μία ζωή και ελέγχει αν οι ζωές του κύβου είναι 0. Αν οι ζωές είναι 0, τότε καλείται η βοηθητική συνάρτηση **PlayBlockBreak( )**, η οποία κάνει τον ήχο για το σπάσιμο ενός κύβου και στη συνέχεια καλείται η συνάρτηση **breakToLittleBlocks( )** της κλάσης (script) Cubes, η οποία εμφανίζει 5 μικρά κυβάκια στη θέση του μεγάλου που σπάει. Στις γραμμές 286-292 αποφασίζεται με πιθανότητα 20% αν το κυβάκι που σπάει θα πετάξει στο πάτωμα ένα σφυρί ώστε ο παίκτης να μπορεί να το μαζέψει. Τέλος, καλείται η μέθοδος Destroy( ) της Unity με την οποία καταστρέφουμε τον κύβο.

- Στις γραμμές 305-334 υλοποιείται η αλλαγή μεταξύ 2 καμερών, του παίκτη και μίας εξωτερικής, δηλαδή το bonus ερώτημα 3.
- Στις γραμμές 337-368 υλοποιείται το τέλος παιχνιδιού είτε με το κουμπί X πρόωρα είτε με το κουμπί E αν ο παίκτης είναι στο τελευταίο επίπεδο. Το παιχνίδι φορτώνει την σκηνή GameOver όπου αναγράφεται το τελικό σκορ.
- Στις γραμμές 403-425 είναι οι συναρτήσεις που χειρίζονται τους colliders των μαύρων κουτιών (bonus ερώτημα 2). Όταν ο χρήστης μπαίνει στο μαύρο κουτί, καλείται η συνάρτηση **PlayTeleport()** η οποία παίζει τον ήχο της τηλεμεταφοράς και η φωτεινή πηγή που ακολουθεί τον παίκτη αυξάνει την ένταση στο 25 ώστε να υπάρχει ένα εφέ λάμψης. Έπειτα η θέση του παίκτη αλλάζει στην θέση του μαύρου κουτιού του παραπάνω επιπέδου. Όταν ο χρήστης βγαίνει από το μαύρο κουτί, η φωτεινή πηγή που τον ακολουθεί επανέρχεται στην ένταση 1.

## 2. Cubes

Το script “Cubes” στην ουσία είναι μία κλάση που διαχειρίζεται έναν κύβο τη φορά. Περιέχει τα κατάλληλα πεδία που χρειάζεται ο κύβος όπως ζωές, χρώμα και θέση (Vector3), καθώς κρατάει και το ίδιο το αντικείμενο (GameObject) του κύβου που δημιουργεί σαν πεδίο.

- Στις γραμμές 21-91 έχουμε μία μέθοδο την οποία καλεί το Script «Game» στην αρχή (συνάρτηση Start( )), η οποία δημιουργεί, χρωματίζει ανάλογα με το όρισμα και επιστρέφει ένα αντικείμενο GameObject το οποίο είναι ένας κύβος του maze μας, Αρχικοποιεί επίσης τα βασικά πεδία του κύβου όπως ζωές, position, color.
- Στις γραμμές 93-95 είναι μια απλή συνάρτηση η οποία μειώνει τη ζωή ενός κύβου κατά 1 και χρησιμοποιείται όταν ο παίκτης χτυπάει έναν κύβο με το σφυρί, μέσω του Script “Game”, μέσω της μεθόδου Update( ).
- Στις γραμμές 102-117 υπάρχουν κάποιοι getters ώστε να μπορούμε να αποκτήσουμε πρόσβαση στα πεδία του κύβου από άλλες κλάσεις.
- Στις γραμμές 119-129 υλοποιείται το σπάσιμο ενός κύβου σε 5 μικρότερους κύβους του ίδιου χρώματος, οι οποίοι με την τελευταία εντολή της μεθόδου καταστρέφονται μετά από 5 δευτερόλεπτα.
- Τέλος, στις γραμμές 131-142 υλοποιείται μία μέθοδος με αρκετές εντολές που αφορούν τα materials στην Unity, η οποία επιστρέφει ένα τροποποιημένο material που το δίνουμε σαν material στους κύβους έτσι ώστε να μπορούμε να τους κάνουμε μερικώς διαφανείς για το 3<sup>ο</sup> bonus ερώτημα (Το default material δεν μπορούσε να προσαρμοστεί ώστε να περιέχει διαφάνεια).

### 3. animController

Το Script “animController” λειτουργεί ως χειριστής του animation του pickaxe. Στην μέθοδο Start( ) απλά θέτει ένα αντικείμενο Animator ως τον Animator του pickaxe και στην Update( ) παίζει το animation κάθε φορά που ο χρήστης πατάει αριστερό κλικ ή το πλήκτρο H και έχει διαθέσιμα pickaxes.

### 4. keyboardorbit

Το Script “keyboardorbit” χειρίζεται την εξωτερική κάμερα που ενεργοποιείται όταν ο παίκτης πατήσει το κουμπί C ώστε να δει το maze εξωτερικά. Αυτό το script είναι attached σε εκείνη την κάμερα και βοηθάει τον χρήστη να μετακινηθεί στον χώρο χρησιμοποιώντας το ποντίκι του.

### 5. PickUp

Το Script “PickUp” είναι attached στον “FPSController” και δουλειά του είναι να «μαζέψει» ένα pickaxe που έχει πέσει τυχαία από κουτί όταν ο παίκτης είναι κοντά και το κοιτάζει (με τη χρήση raycast). Αποτελείται από τη μέθοδο Update( ) η οποία και ενεργοποιείται αν ο παίκτης δει pickaxe που έχει πέσει από κουτί, τότε το εξαφανίζει και αυξάνει τα διαθέσιμα pickaxes κατά 1.



## 6. GameOver

Το Script “GameOver” ενεργοποιείται όταν αλλάξουμε Scene στο “GameOver” scene και στην Update( ) του απλά αλλάζει το κείμενο του Script προβάλλοντας το σωστό τελικό σκορ στον χρήστη και παρέχει τη δυνατότητα να κλείσει το παιχνίδι πατώντας το πλήκτρο X.

## ΒΙΒΛΙΟΓΡΑΦΙΑ

Αρχικά, βασιστήκαμε πάνω σε κάποια πράγματα από τα αρχεία του φροντιστηρίου (κάποια τα αναφέρουμε παραπάνω).

Έπειτα, χρήσιμα ήταν τα παρακάτω βίντεο/posts:

[https://www.youtube.com/watch?v=uqTUjnb9Wc&fbclid=IwAR2J\\_aiE67OIKRrGHqKsc3PqedLwmadm5Gvb6PqKY-ssn3xHleMcjiJ5Bfo](https://www.youtube.com/watch?v=uqTUjnb9Wc&fbclid=IwAR2J_aiE67OIKRrGHqKsc3PqedLwmadm5Gvb6PqKY-ssn3xHleMcjiJ5Bfo)

<https://www.youtube.com/watch?v=JnbDxG04i7c&fbclid=IwAR3Czqi58tGskBhLCD0DsQB8ueeltO-2zyWhCxZj4kbJGyJ5Q0lsaRAbG3E>

[https://forum.unity.com/threads/access-rendering-mode-var-on-standard-shader-via-scripting.287002/?\\_ga=2.105856003.1854611508.1578444074-404423678.1578444074#post-1961025](https://forum.unity.com/threads/access-rendering-mode-var-on-standard-shader-via-scripting.287002/?_ga=2.105856003.1854611508.1578444074-404423678.1578444074#post-1961025)

<https://forum.unity.com/threads/animation-wont-play-on-keypress.344350/>

καθώς και η προσωπική εμπειρία πάνω στην Unity και την C#.