

2η Προγραμματιστική Άσκηση

Γκαβαρδίνας Όθωνας, AM: 2620
Μπουρλή Στυλιανή, AM: 2774

Σκοπός της άσκησης

Στόχος ήταν η εξοικείωση με τη χρήση πλατφορμών γραφικών, όπως η Unity3D. Η Unity 3D παρέχει μία μεγάλη ποικιλία από εργαλεία για ανάπτυξη παιχνιδιών και άλλων αλληλεπιδραστικών εφαρμογών. Πιο συγκεκριμένα, σκοπός ήταν η δημιουργία ενός απλού παιχνιδιού «χτισίματος 3D σκηνής» τύπου Minecraft.

Γλώσσα: C#

Έκδοση Unity: 2018.2.18f1 Personal(64bit) Windows

Υλοποίηση

1.

Αρχικό παράθυρο

Το **scene Menu** αναφέρεται στο παράθυρο που ανοίγει με την έναρξη του προγράμματος. Αυτό περιέχει μία κάμερα που δείχνει στο παράθυρο και ένα αντικείμενο **Canvas**. Σ' αυτό υπάρχει:

- Ένα **Panel** με όνομα **Background**, που αφορά το φόντο του παραθύρου και είναι χρώματος μπλε
- Ένα **InputField** για την είσοδο από το χρήστη του μεγέθους **N**, όπου είναι το μέγεθος του πλέγματος (νοητό 3D πλέγμα $N \times N \times N$)
- Ένα **Button** με όνομα **PlayButton**, για την έναρξη του παιχνιδιού.

Στο **scene Menu** χρησιμοποιούνται, επίσης, δύο **script**:

- Το **TheScene**, που περιέχει τη συνάρτηση **setSize()**, η οποία χρησιμοποιείται για την αποθήκευση του **N** (**sceneSize**), το οποίο δίνεται στο **InputField**. Το **sceneSize** έχει οριστεί ως **static** για να μπορεί να χρησιμοποιηθεί και στο επόμενο **scene**.

- Το **Menu**, που περιέχει τη συνάρτηση **PlayGame()**, η οποία εκτελεί την εναλλαγή σκηνής με το πάτημα του κουμπιού **PlayButton**. Για να πραγματοποιηθεί αυτό θα πρέπει το **sceneSize** που θα δοθεί να είναι μεγαλύτερο του 1.

Πάτωμα πλέγματος

Στο **scene Game** υλοποιείται το πάτωμα του πλέγματος. Πιο συγκεκριμένα, υπάρχει ένα κενό (empty) αντικείμενο **GameObject**, το οποίο περιέχει το **script CubeCreator**, που δημιουργεί το πλέγμα και το **script TheScene**, που περιέχει το μέγεθος **N** του πλέγματος.

Στο **script CubeCreator** δημιουργούνται κύβοι διάστασης 2x2x2 και τοποθετούνται συνεχόμενα, ώστε να γεμίσουν μια επιφάνεια διάστασης NxNxN. Το χρώμα για τον κάθε κύβο επιλέγεται τυχαία μέσω της συνάρτησης **setRandomColour()**. Τα διαθέσιμα χρώματα είναι κόκκινο, κίτρινο, πράσινο, γαλάζιο και μπλε. Ως εξαίρεση ο κεντρικός κύβος έχει χρώμα magenta.

Για τον κύβο δημιουργήθηκε ένα **prefab cube.*

Τοποθέτηση του παίκτη

Ο παίκτης ορίζεται ως **First Person**, δηλαδή μία κάμερα που τοποθετείται αρχικά πάνω στον κεντρικό κύβο του πατώματος του πλέγματος. Η αρχική τοποθέτηση του παίκτη πραγματοποιείται στο **script CubeCreator**.

2.

Ο παίκτης κατεβαίνει

Στον **FirstPerson** υπάρχει το **script CameraHeight** που αφορά την κίνηση του παίκτη στα διάφορα επίπεδα. Πιο συγκεκριμένα, υπάρχει ο **CharacterController**, ο οποίος ελέγχει εάν ο παίκτης ακουμπάει το έδαφος κάποιου επιπέδου. Επίσης, στην περίπτωση που ο παίκτης βρεθεί σε ύψος μικρότερο από αυτό που ήταν προηγουμένως, υπολογίζονται ο αριθμός των επιπέδων που έπεσε ο παίκτης, και για κάθε ένα εκτός του πρώτου αφαιρούνται 10 βαθμοί από το score. Η μείωση των βαθμών υπολογίζεται στην συνάρτηση **DecreaseScore()**, στην οποία ελέγχεται επιπλέον η περίπτωση που το score μετά τη μείωση πρόκειται να πέσει κάτω από το 0. Σ' αυτήν την περίπτωση οι ζωές του παίκτη μειώνονται κατά μία και το score του γίνεται 0.

Ο παίκτης πέφτει στο κενό

Στο **script CameraHeight** ελέγχεται επίσης η περίπτωση που ο παίκτης πέφτει στο κενό. Πιο συγκεκριμένα, αν το ύψος του παίκτη μειωθεί περισσότερο από -40 οι ζωές του παίκτη μειώνονται κατά μία, και αν επαρκούν το παιχνίδι συνεχίζεται με την επαναφορά του παίκτη στον κεντρικό κύβο του πατώματος πλέγματος.

Ο παίκτης ανεβαίνει επίπεδο

Στο **script CameraHeight**, με τη βοήθεια του **CharacterController** και ύψους που βρίσκεται ο παίκτης ελέγχεται αν ο παίκτης ακουμπά σε επίπεδο μεγαλύτερου ύψους από αυτό που βρισκόταν προηγουμένως. Σε αυτήν την περίπτωση ο παίκτης κερδίζει 10 πόντους.

Νοητό πλέγμα

Στο κενό αντικείμενο **GameObject**, βρίσκεται το **script BoarderMaker**, το οποίο δημιουργεί ένα νοητό πλέγμα μέσα στο οποίο περιορίζεται η κίνηση του παίκτη. Σ' αυτό υπάρχει η συνάρτηση **CreateBoarders()**, η οποία περικλείει με νοητά φράγματα, τον NxNxN χώρο, όπου επιτρέπεται να κινείται ο χρήστης.

Για το όριο δημιουργήθηκε ένα **prefab boarder, που παραμετροποιείται.*

3.

Φωτισμός σκηνής

Στο **script BoarderMaker**, στη συνάρτηση **CreateBoarders()**, δημιουργούνται δύο αντικείμενα τύπου **SpotLight**, τα οποία τοποθετούνται αντιδιαμετρικά στη σκηνή σε ύψος N.

Για τη φωτεινή πηγή δημιουργήθηκε ένα **prefab spotlight, που παραμετροποιείται.*

Παράθυρο του παίκτη

Στο παράθυρο του παίκτη κατά τη διάρκεια του παιχνιδιού εμφανίζονται επιπλέον οι βαθμοί του, οι ζωές του και το απόθεμα κύβων και κυλίνδρων που διαθέτει. Για αυτή τη λειτουργία, στο **FPSController** υπάρχει ένα αντικείμενο τύπου **Canvas**, με όνομα **LiveUI**, στο οποίο περιέχονται τα labels και τα text για τα παραπάνω στοιχεία. Τα text μεταβάλλονται αναλόγως σε διάφορα σημεία του κώδικα. Όσον αφορά τις ζωές, για τις 4 πρώτες χρησιμοποιήθηκε μία εικόνα με περιπτώσεις άδειων και γεμάτων καρδιών. Η εικόνα χωρίστηκε σε 5 κομμάτια για κάθε περίπτωση, τα οποία αποθηκεύτηκαν σε ένα πίνακα. Για τις υπόλοιπες, προστέθηκε το σύμβολο '+', καθώς και ένα label. Τα δεδομένα που αφορούν την αυξομείωση των ζώων, τα διαχειρίζεται το **script LivesManager**, που βρίσκεται στο αντικείμενο-εικόνα **Lives**.

Οι εικόνες βρίσκονται στον φάκελο **Graphics.*

4.

Συλλογή αποθέματος

Στον **FPSController**, υπάρχει το **script RayCasting** το οποίο αφορά την αλληλεπίδραση του παίκτη με τα αντικείμενα. Όταν ο παίκτης πατήσει το πλήκτρο <p> και το αντικείμενο που κοιτάζει είναι κύβος, και το χρώμα του δεν είναι μπλε ή magenta, παίρνει ένα απόθεμα από κύβους ή κυλίνδρους. Κάθε κύβος έχει το δικό του απόθεμα, και το χρώμα του κύβου αλλάζει ανάλογα με το απόθεμα που του απομένει. Ο γαλάζιος κύβος όταν δώσει το απόθεμά του εξαφανίζεται, χωρίς να πέσουν τα αντικείμενα που βρίσκονται πάνω από αυτόν. Κάθε φορά που ο παίκτης αυξάνει το απόθεμά του, οι βαθμοί του μειώνονται κατά 5, με τη συνάρτηση **decreaseScore()**.

5.

Τοποθέτηση κύβων

Στον **FPSController**, στο **script RayCasting**, υλοποιείται και η περίπτωση τοποθέτησης κύβου με το πάτημα του πλήκτρου . Συγκεκριμένα, όταν ο παίκτης πατήσει το πλήκτρο και έχει απόθεμα κύβων, τοποθετείται ένας κύβος τυχαίου χρώματος που επιλέγεται με τη συνάρτηση **setRandomColour()** στο σημείο που κοιτάζει ο παίκτης και οι βαθμοί του παίκτη αυξάνονται κατά 10. Αν σε αυτό το σημείο υπάρχει ήδη κάποιο αντικείμενο, ο καινούριος κύβος τοποθετείται στο μεγαλύτερο δυνατό ύψος (έως N). Αυτό επιτυγχάνεται μέσω της συνάρτησης **cubeFindHeight()**. Η συνάρτηση αυτή παίρνει ως όρισμα τις συντεταγμένες που κοιτάζει ο χρήστης, ψάχνει σε ένα λεξικό για τα ύψη των αντικειμένων που έχουν τοποθετηθεί σ' αυτές τις συντεταγμένες και αν βρει κάποια κενή θέση τοποθετεί εκεί το αντικείμενο. Το λεξικό αυτό ονομάζεται **objectsPositions** και δημιουργείται στο **CubeCreator**, όπου αρχικοποιείται με τις συντεταγμένες και τα ύψη των κύβων που αποτελούν το πάτωμα.

6.

Τοποθέτηση κυλίνδρων

Στον **FPSController**, στο **script RayCasting**, υλοποιείται και η περίπτωση τοποθέτησης κυλίνδρου με το πάτημα του πλήκτρου <c>. Συγκεκριμένα, όταν ο παίκτης πατήσει το πλήκτρο και έχει απόθεμα κυλίνδρων, τοποθετείται ένας κύλινδρος στο σημείο που κοιτάζει ο παίκτης και οι βαθμοί του παίκτη αυξάνονται κατά 20. Αν σε αυτό το σημείο υπάρχει ήδη κάποιο αντικείμενο, ο καινούριος κύλινδρος τοποθετείται στο μεγαλύτερο δυνατό ύψος (έως N). Αυτό επιτυγχάνεται μέσω της συνάρτησης **cylinderFindHeight()**. Η συνάρτηση αυτή παίρνει ως όρισμα τις συντεταγμένες που κοιτάζει ο χρήστης, ψάχνει σε ένα λεξικό για τα ύψη των αντικειμένων που έχουν τοποθετηθεί σ' αυτές τις συντεταγμένες και αν βρει κάποια κενή θέση τοποθετεί εκεί το αντικείμενο. Ο κύλινδρος καταλαμβάνει δύο θέσεις ύψους στη λίστα, στο λεξικό.

7.

Άφιξη στο επίπεδο N

Όταν ο παίκτης ανέβει πρώτη φορά στο επίπεδο N, στο **script cameraHeight**, αυξάνονται οι βαθμοί του κατά 100 και παίρνει μία ζωή. Το αν έφτασε στο επίπεδο N, ελέγχεται μέσω της μεταβλητής **level** που ορίζεται στο **script CubeCreator** και του **CharacterController** που ελέγχει αν ο παίκτης ακουμπά στο έδαφος.

Όταν ο παίκτης ανέβει πρώτη φορά στο επίπεδο N-1, δημιουργούνται από το **script BordersMaker** επιπλέον όρια και φωτεινές πηγές σε διπλάσιο ύψος από το προηγούμενο ώστε να καλύψουν τις ανάγκες των επόμενων επιπέδων. Αυτό επιτυγχάνεται με την μεταβλητή **boarderLevel**.

Τα παραπάνω εξασφαλίζουν ότι το παιχνίδι μπορεί να συνεχιστεί και για περισσότερα από N επίπεδα, αρκεί ο παίκτης να έχει ζωές.

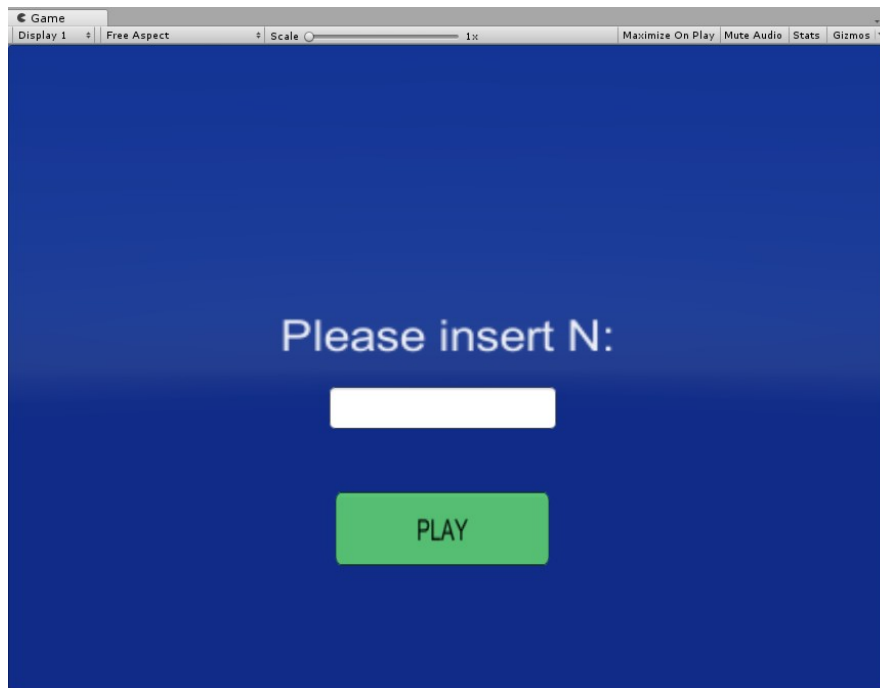
Επιπλέον λειτουργία – Τερματισμός παιχνιδιού

Στην περίπτωση που ο παίκτης χάσει όλες τις ζωές του, η σκηνή μεταφέρεται στην σκηνή **GameOver**, η οποία περιέχει ένα παράθυρο με τους βαθμούς που κατάφερε να συλλέξει ο παίκτης.

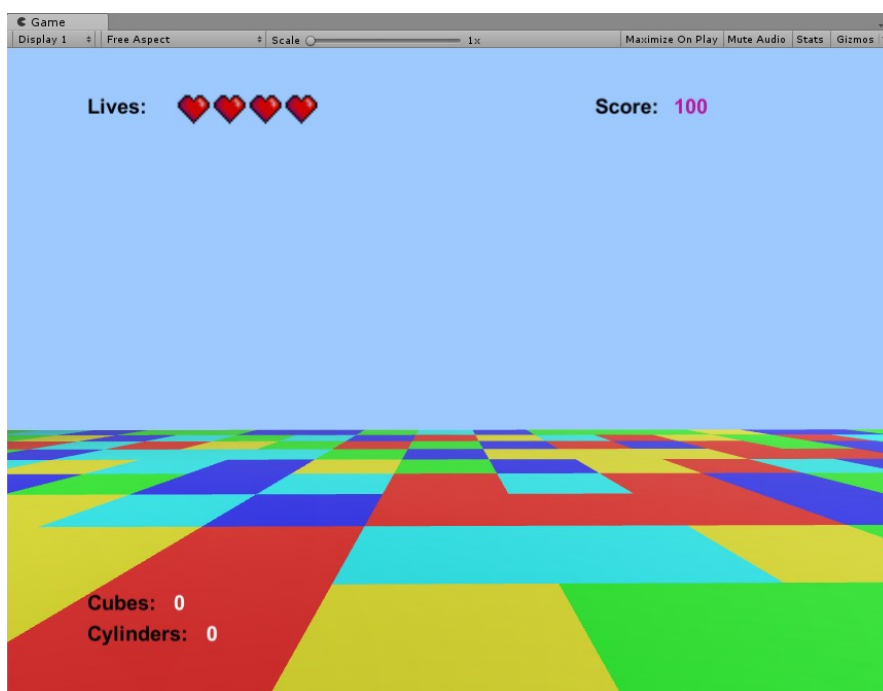
Παράδειγμα Εκτέλεσης Παιχνιδιού

Είσοδος στο παιχνίδι

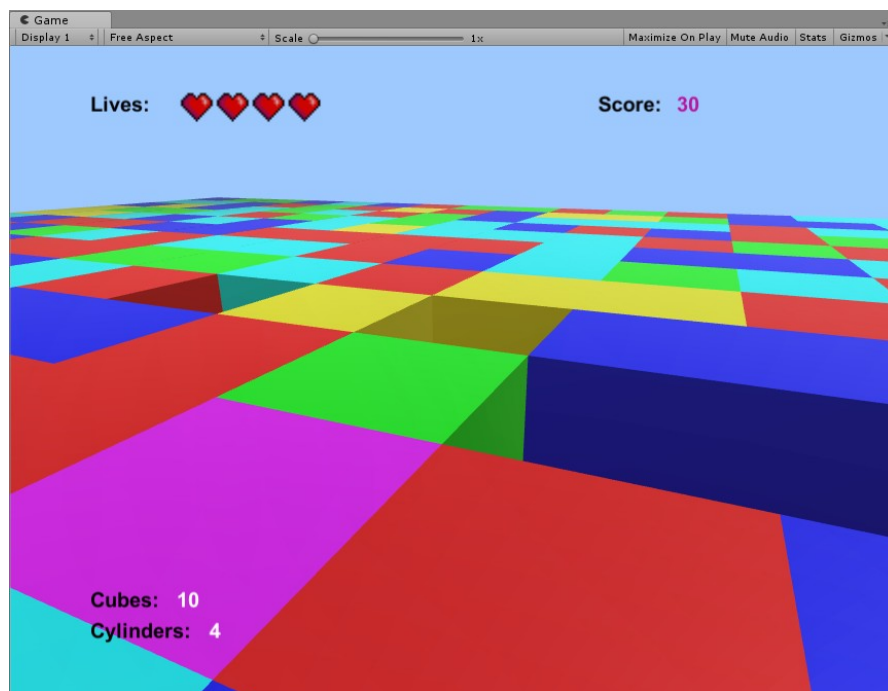
Με την είσοδο στο παιχνίδι εμφανίζεται ένα παράθυρο στο οποίο πρέπει ο παίκτης να εισάγει το N και να πατήσει το κουμπί PLAY.



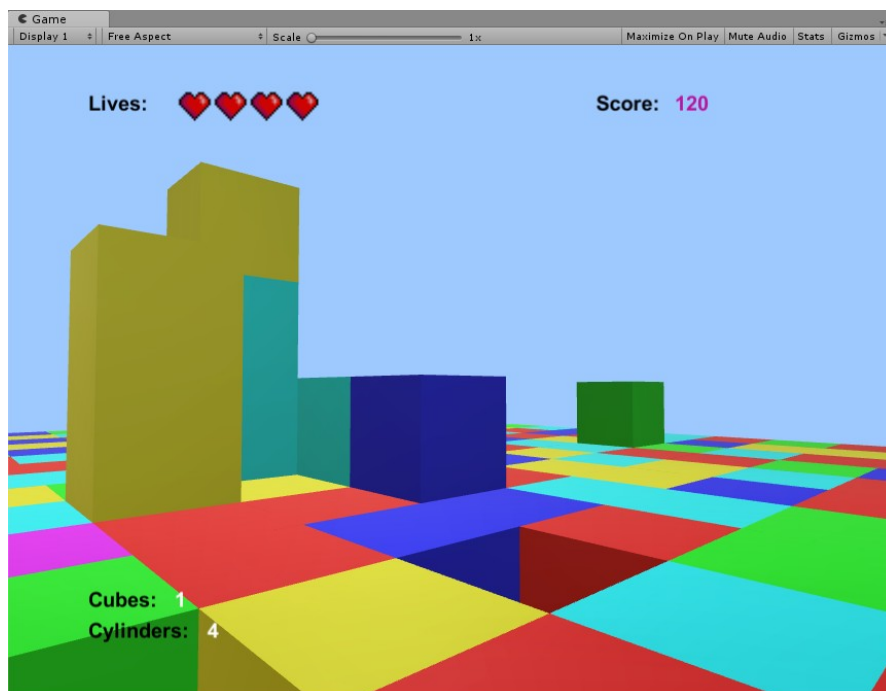
Στη συνέχεια, ο παίκτης μπορεί να κινηθεί στο χώρο με τα πλήκτρα WASD ή τα βελάκια σε συνδυασμό με το ποντίκι. Επίσης, μπορεί να πηδήξει με το πλήκτρο space.



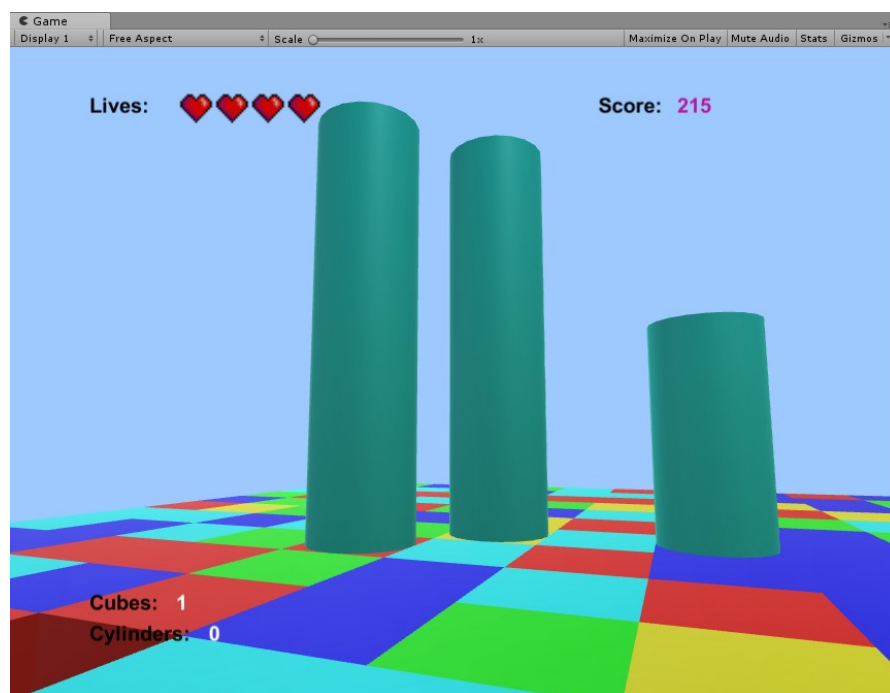
Με το πλήκτρο <p> συλλέγει κύβους και κυλίνδρους.



Με το πλήκτρο τοποθετεί κύβους.



Με το πλήκτρο <c> τοποθετεί κυλίνδρους.



Όταν ο παίκτης χάσει όλες τις ζωές.

