

Μάθημα: Γραφικά Υπολογιστών και Συστήματα Αλληλεπίδρασης

Ονοματεπώνυμο, ΑΜ:

Παναγιώτης Κουζουγλίδης, 2276

Αγγελική Αλεξανδρή, 2204

Project: 3D Maze

Έτος: 2015-2016

Ξεκίνημα παιχνιδιού:

Το παιχνίδι ξεκινάει με την κάμερα να δείχνει πάνω από τον λαβύρινθο. Όταν ο χρήστης πατήσει το "R" τότε η κάμερα αρχίζει και περιστρέφεται γύρω από τον λαβύρινθο. Και τις δύο φάσεις που προαναφέρθηκαν ο παίκτης μπορεί να κινηθεί κανονικά απλά η κάμερα είναι από πάνω και πιθανώς να μην φαίνεται καν ο κύλινδρος. Όταν ο χρήστης πατήσει το "V" η κάμερα μεταφέρεται μέσα στον λαβύρινθο κοιτώντας εκεί που την άφησε ο χρήστης

Controls:

"W" πηγαίνει μπροστά,

"S" πηγαίνει πίσω

"H" καταστρέφει το τετραγωνάκι που κοιτάει

"X" τερματίζει το παιχνίδι και το score γίνεται 0 εφόσον ο χρήστης δεν έχει φτάσει στο επίπεδο L

"Spacebar" ανεβαίνει επίπεδο εφόσον από πάνω του δεν υπάρχει block. Εφόσον βρίσκεται στο επίπεδο L τότε το παιχνίδι τερματίζει και εμφανίζεται το τελικό score.

"E" εφόσον ο παίκτης βρίσκεται στο επίπεδο L το παιχνίδι τερματίζει και εμφανίζεται το τελικό score.

"V" η κάμερα μεταφέρεται εντός του λαβυρίνθου εκεί που την άφησε ο χρήστης. Άμα το πατήσει την στιγμή που η κάμερα είναι ήδη μέσα στον λαβύρινθο τότε η κάμερα μεταφέρεται έξω από τον λαβύρινθο και παραμένει ακίνητη.

"R" εφόσον η κάμερα κοιτάει έξω από τον λαβύρινθο, η κάμερα αρχίζει και περιστρέφεται γύρω από τον λαβύρινθο. Επίσης όταν το παιχνίδι τερματίσει μπορεί να το πατήσει ώστε να γίνει restart το παιχνίδι.

"O" αυξάνει την διαφάνεια των block ώστε να φαίνεται το εσωτερικό τους

"P" μειώνει την διαφάνεια των block

"Esc" κλείνει τελείως το παράθυρο του παιχνιδιού

Κίνηση κάμερας

Ο παίκτης μπορεί να κουνήσει το ποντίκι αριστερά ή δεξιά ώστε να περιστραφεί καταλλήλως η κάμερα. Δεν καταφέραμε να την κάνουμε την κάμερα να κουνιέται πάνω και κάτω και γι αυτό κάναμε να εμφανίζεται κείμενο που θα λέει στον χρήστη άμα υπάρχει block από πάνω (άμα υπάρχει μπορεί να πατήσει “G” για να το καταστρέψει), και άμα υπάρχει “τρύπα” (ώστε να πέσει ένα ή παραπάνω επίπεδα) στο επόμενο block στην κατεύθυνση που κοιτάει η κάμερα.

Ζωγραφισμός Λαβυρίνθου

Κάθε block αρχικοποιείται με κάποιες συντεταγμένες και έπειτα ζωγραφίζεται στην κατάλληλη θέση. Επίσης για κάθε επιφάνεια ορίζουμε τα κατάλληλα normals για τον φωτισμό

Τοίχοι και πόρτες

Οι πόρτες δεν επηρεάζονται από την διαφάνεια καθώς δεν είναι ανάγκη να μπορούμε να δούμε μέσα απο αυτές. Για κάθε μεριά του τοίχου και για κάθε μεριά των πορτών ορίζονται τα κατάλληλα normals για τον φωτισμό.

Παρατήρηση: Δεν καταφέραμε να κάνουμε το collision για τις πόρτες

Τηλεμεταφορά

Εφόσον ο χρήστης μπει σε κάποιο τετράγωνο τηλεμεταφοράς τότε μεταφέρεται στο κέντρο του τετραγώνου του άλλου τετραγώνου τηλεμεταφοράς που υπάρχει στο επίπεδο που βρίσκεται ο παίκτης. Το lookAngle κατά την τηλεμεταφορά δεν αλλάζει.

Διαφάνεια

Τα controls της διαφάνειας περιγράφονται στα controls που είναι πιο πάνω
Minimum τιμή διαφάνειας: 0.1

Φωτισμός

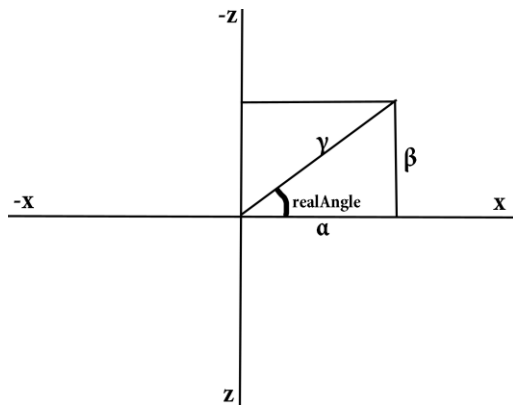
Βάλαμε μια πηγή φωτός που φωτίζει όλο τον λαβύρινθο αλλά δεν καταφέραμε να κάνουμε κάποιο φως που ακολουθεί τον παίκτη (φακό δηλαδή).

Score

Κατά την διάρκεια του παιχνιδιού το score του παίκτη φαίνεται πάνω δεξιά. Όταν το παιχνίδι τερματίσει για οποιονδήποτε λόγο τότε εμφανίζεται ένα ορθογώνιο σαν παράθυρο όπου μέσα εμφανίζει το τελικό score.

Κίνηση στον χώρο

Ο παίκτης μπορεί να κουνηθεί σε όποια κατεύθυνση και αν κοιτάει. Η λογική είναι ότι με βάση το lookAngle υπολογίζεται το realAngle (όπως φαίνεται στο παρακάτω σχήμα). Έπειτα αυξάνουμε το μήκος της πλευράς “γ” κατά πολύ λίγο και τελικώς υπολογίζουμε τα καινούρια μήκη των “α” και “β” που είναι οι καινούριες συντεταγμένες x,z.



Παρατηρήσεις:

-Δεν δώσαμε και τόσο έμφαση στο διάβασμα του maze από αρχείο όσον αφορά την εγκυρότητα ή μη της εισόδου. Είναι κάτι που έχουμε κάνει πάρα πολλές φορές και δεν νομίζουμε ότι αυτό είναι η ουσία της άσκησης.

-Δεν καταφέραμε να το τρέξουμε στους υπολογιστές της σχολής γιατί έβγαζε διάφορα σφάλματα στο compile.

Στο turnin παραδόθηκαν και τα κατάλληλα .png για τα textures.