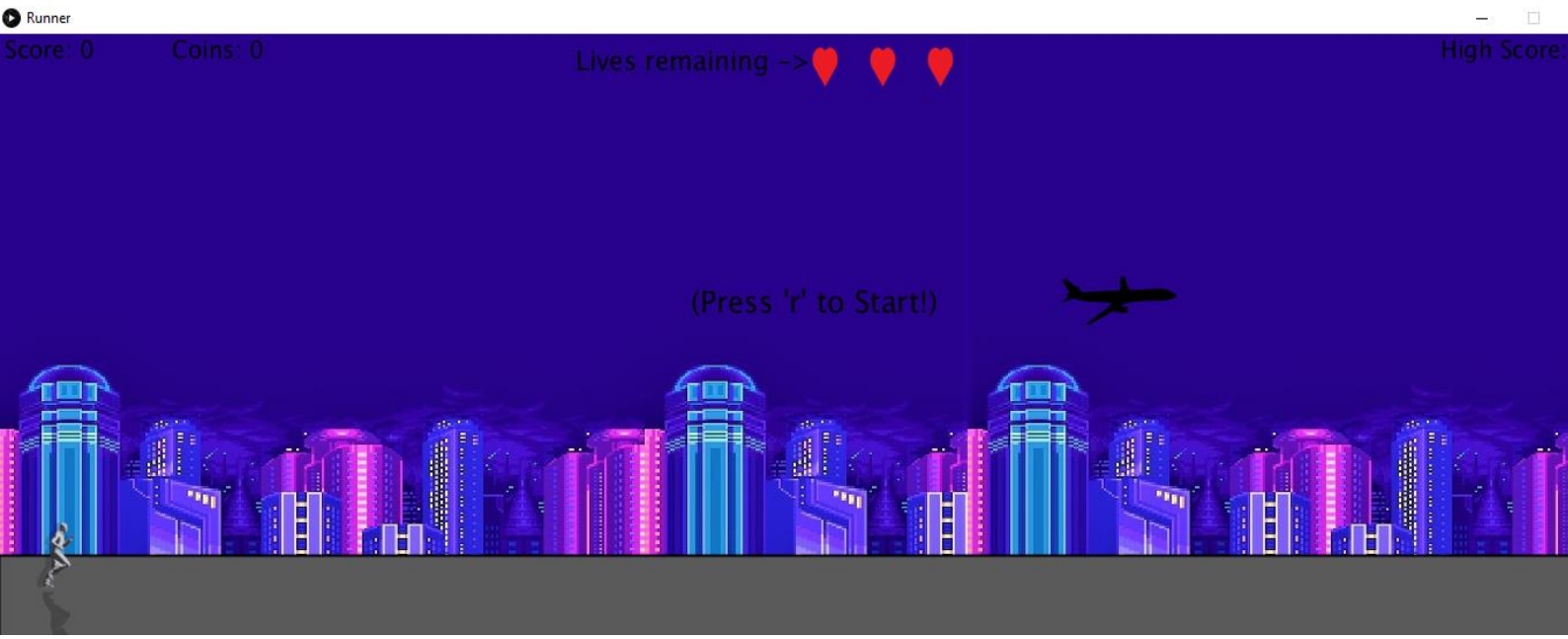
2η εργασία_Κατασκευή παιχνιδιού με Processing

Μέλη ομάδας:

- ΦΙΛΙΠΠΟΣ ΔΟΥΡΑΧΑΛΗΣ / 3170045
- ΙΩΑΝΝΗΣ ΚΟΤΤΑΣ / 3170220
- ΔΗΜΗΤΡΗΣ ΜΠΑΣΤΑΣ / 3130139
- ΚΑΡΑΜΗΤΡΟΥ ΑΝΝΑ-ΜΑΡΙΑ / 3160053

Μια πρώτη ματιά



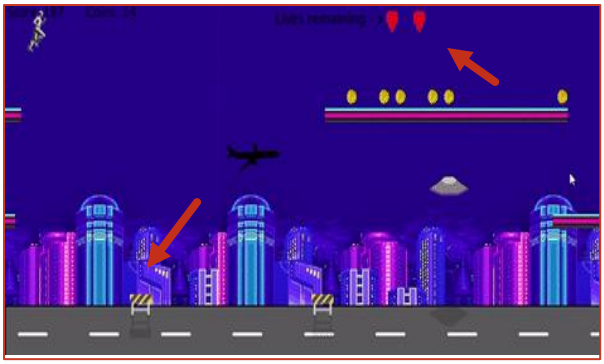
RUNNER

Περιεχόμενα

Γενικές οδηγίες παιχνιδιού

ΕΙΚΟΝΑ 1

Το παιχνίδι κινείται γύρω από ένα άνθρωπο (runner - ΕΙΚΟΝΑ 1), ο οποίος συνεχώς κινείται δεξιά και έχει την δυνατότητα να πηδήξει και να ανέβει σε κάποια πλατφόρμα. Όσο μεγαλύτερο διάστημα διανύσει, τόσο μεγαλύτερο highScore πετυχαίνει. Στην διαδρομή εμφανίζονται συνεχώς εμπόδια και η ταχύτητα όλο και αυξάνεται για περισσότερη. Για κάθε εμπόδιο που δεν καταφέρνει να αποφύγει χάνει μία απο τις διαθέσιμες ζωές (ΕΙΚΟΝΑ 2). Επιπρόσθετα μέσα στο παιχνίδι εκτός από εμπόδια υπάρχουν και βοηθητικά αντικείμενα για τον runner των εικόνων. Υπάρχουν τα νομίσματα, τα οποία μπορεί να μαζεύει και όταν φτάνει στα πενήντα θα κερδίζει μία ακόμη ζωή.Υπάρχει ένα σαλιγκάρι, το οποίο μειώνει λίγο την ταχύτητα, αλλά και επιπλέον ζωές. Σε παρακάτω ενότητα θα επεξηγήσουμε κάθε κλάση ξεχωριστά.



ΕΙΚΟΝΑ 2

Ο χρήστης έχει την δυνατότητα να κάνει δύο κινήσεις, να πηδήξει πατώντας ‘space’ μία φορά για μικρό άλμα , δύο φορές για μεγαλύτερο άλμα και πατώντας παρατεταμένα την δεύτερη φορά το άλμα φτάνει την μέγιστη δυνατή απόσταση. Επιπλέον όταν είναι πάνω σε μία πλατφόρμα έχει την δυνατότητα να

πατήσει ‘s’ και να κατέβει στην απο κάτω. Τέλος όταν χάσει όλες τις ζωές το παιχνίδι τερματίζει και πατώντας ‘r’ έχει την επιλογή να ξαναξεκινήσει από την αρχή.

Βιβλιοθήκη

Για προσθήκη ηχητικών εφέ, εγκαταστήσαμε μια βιβλιοθήκη για sounds με αποτέλεσμα σε κάθε σύγκρουση να έχουμε και ένα διαφορετικό ηχητικό εφέ. Επιπλεόν βάλαμε και μια μουσική στο background, η οποία επαναλαμβάνεται συνεχώς.

Συνεισφορά κάθε μέλους στην ομάδα

Αρχικά μετά από έρευνα και όλοι μαζί αποφασίσαμε τι παιχίδι θα κατασκευάσουμε και ποια θα είναι τα βασικά χαρακτηριστικά του.Η πρώτη σκέψη που είχαμε ήταν να υλοποιήσουμε ένα mini game σαν το δεινοσαυράκι της Google(πηγή). Συνεπώς για αρχή κατασκευάσμαε ένα αρχικό template πάνω και στο οποίο βασιστήκαμε στα διάφορα στάδια της εργασίας. Γενικά εργαστήκαμε ομαδικά ακολουθώντας τα παρακάτω στάδια.

1. Πραγματοποίηση κλήσης για συζήτηση με τι θα ασχοληθούμε και ανταλλαγή ιδεών.
2. Προσπάθεια υλοποίησης ενός κομματιού της εργασίας από τον καθένα ξεχωριστά.
3. Επανάληψη κλήσης για να συζητήσουμε τις προσπάθειες μας.
4. Απόφαση για την καλύτερη έκδοση,δηλαδή την πιο αποδοτική και την πιο όμορφη αισθητικά.
5. Συνένωση συγκεκριμένου κομματιού με τον υπόλοιπο κώδικα.

Επεξήγηση κώδικα

Ο κώδικας χωρίζεται στις παρακάτω κλάσεις:

Runner → Αποτελεί την main κλάση του παιχνιδιού μας, όπου γίνεται αρχικοποίηση των ήχων και των εικόνων που θα χρησιμοποιηθούν και εισαγωγή εικόνας background. Εμφανίζει διάφορα text στην οθόνη, υπολοιπόμενες ζωές του παίκτη, το HighScore, το score την κάθε στιγμή, τα coins που μαζεύει, αλλά και μήνυμα όταν πεθαίνει, το οποίο αναφέρει τον τρόπο που έχασε την τελευταία ζωή του.

- Collisions (): έλεγχος για επαφή με πλατφόρμα και αν καταφέρνει να ανέβει σε αυτή, έλεγχος αν συγκρούεται με κάποιο εμπόδιο και αν συγκρουστεί και δεν έχει άλλη ζωή ενημέρωση κατάλληλης μεταβλητής.
- Update (): κίνηση αντικειμένων που ήδη υπάρχουν και ενημέρωση μεταβλητών για κλήση συναρτήσεων που θα προσθέτουν πλατφόρμες, εμπόδια και έδαφος. Επίσης σταδιακή αύξηση ταχύτητας εμποδίων.
- addObstacle(): σύμφωνα με μία random μεταβλητή προσθήκη εμποδίου.
- createPlatform(): δημιουργία πλατφορμών, με τυχαία θέση.

- destroyObjects(): όταν κάποιο αντκείμενο φεύγει από οθόνη τότε διαγραφή από την αντίστοιχη λίστα.
- keyPressed(): ανάλογα το κουμπί που πατιέται, ανάλογη κίνηση παίκτη.

gameObject → Αποτελεί μία abstract κλάση, στην οποία γίνεται η υλοποίηση των παρακάτω μεθόδων οι οποίες override από τις κλάσεις που την κάνουν extends.

- collision (): συγκρίνοντας τις θέσεις x, y από 2 αντικείμενα επιστρέφει ανάλογα true ή false.
- Move (): ενημερώνει την θέση x ενός αντικειμένου με συγκεκριμένο speed.
- Draw (): εμφανίζει την εικόνα του αντικειμένου.
  Επιπρόσθετα έχει κάποιες get και set μεθόδους.

Enemy → στον constructor ανάλογα την είσοδο αρχικοποίηση εμποδίου ή αλλιώς εχθρού.

- (override) move(): πρόσθετη λειτουργία της move είναι ότι ένα εμπόδιο κινείται και πάνω-κάτω, ένα άλλο πιο αργά και ένα πιο γρήγορα.
- (override) draw(): προσθήκη σκίασης στους εχθρούς.
- (override) collision (): Αν χάσει απο συγκεκριμένο εχθρό, ενημέρωση μηνύματος.

Ground → απλή κλάση για δημιουργία διακόσμησης εδάφους, δηλαδή άσπρη γραμμή που κάνει το έδαφος να μοιάζει με δρόμο.

Item → Στον constructor τυχαία είσοδος για τυχαία εμφάνιση βοηθητικών αντικειμένων τα οποία ενισχύουν τον παίκτη. Με πενήντα coin ή με καρδιά επιπλέον ζωής, με σαλιγκάρι μείωση ταχύτητας.

- (override) move(): προσθήκη αλλαγή ταχύτητας σε περίπτωση «σύγκρουσης» με σαλιγκάρι.

Platform →Κλάση για δημιουργία πλατφόρμων. Στον constructor αφού γίνει είσοδος μετβλητών, τυχαία προσθήκη των βοηθητικών αντικειμένων.

- (override) draw(): κλήση show() της κλάσης Item για να εμφανιστεί ένα αντικείμενο.
- (override) collision (): ενημέρωση collision έτσι ώστε για κάθε item να προσθέτει την λειτουργία του. Άρα ανάλογα τον τύπο του Item ανάλογη ενέργεια. Αν το αντικείμενο είναι νόμισμα προσθήκη νομίσματος, αν είναι καρδιά, προσθήκη καρδιάς. Επιπρόσθετα προσθήκη ηχητικών εφέ για το κάθε αντικείμενο. Μετά από τις παραπαάνω ενέργειες διαγραφή αντικειμένου από την λίστα.
- onTop(): έλεγχος αν το εισαγώμενο object είναι στις διαστάσεις της πλατφόρμας.

Player → Η συγκεκριμένη κλάση αφορά τον παίκτη και έχει κάποιες αρχικοποιημένες μεταβλητές (πχ default ζωές = 3). Στον κατασκευαστή εισάγει μεταβλητες

που αφορουν την θέση του, την εικόνα του, αλλά και τις διαστάσεις του.

- Jump(): ανάλογα με δύο Boolean μεταβλητές (jump, doubleJump)θέτει πόσο ψηλά θα πηδήξει, έχοντας ανώτατο όριο το δεκαοκτώ.
- (override) move():ενημέρωση μεταβλητών θέσης ανάλογα την κίνηση που υλοποιεί.

- Fall(), MoveDown(): Βοηθητικές μέθοδοι για την κίνηση που κάνει ο παίκτης όταν «πέφτει», ενημέρωση αντίστοιχων μεταβλητών,οι οποίες χρήσιμοποιούνται στην move().
- Lifeloss():Boolean κλάση που αφαιρεί ζωές και αν φτάσουν το μηδέν επιστρέφει false;
- Shadow(): προσθήκη σκίασης στον παίκτη.
- (override) draw(): Σχεδιασμός παίκτη στην οθόνη, ανάλογα με τον πίνακα εικόνων που έχει αρχικοποιηθεί στην Runner.

Περιγραφή προβλήματων

Το κύριο πρόβλημα που αντιμετωπίσαμε ήταν οι συγκρούσεις αντικειμένων. Στο παιχνίδι μας όπως έχει γίνει ήδη κατανοητό ο παίκτης μας τρέχει συνεχόμενα και έρχεται αντιμέτωπος με διάφορα αντικείμενα, είτε εχθρικά, είτε φιλικά. Επιπλέον, άλλη μία δυσκολία ήταν η τυχαία ανάθεση αντικειμένων στην οθόνη, διότι τύχαινε να βγαίνον το ένα πάνω στο άλλο.

Τρόποι επίλυσης

Για το πρώτο πρόβλημα για να κατανοήσουμε πως λειτουργεί μία σύγκρουση, είδαμε ένα template από το παιχίδι μας που αντί για εικόνες βάλαμε σχήματα και όταν ακουμπούσαν δύο μεταξύ τους, τα γεμίζαμε με χρώμα. Κάνοντας διάφορα πειράματα, βγάλαμε το καλύτερο δυνατό αποτέλεσμα.

Πηγές

- https://processing.org/examples/
- https://en.wikipedia.org/wiki/Dinosaur_Game