



Αντικειμενοστρεφής Προγραμματισμός

2022-2023

Εργασία Εξαμήνου

Πατρίκιος Απόστολος Αντώνιος

71347261

Εργασία.....	3
Γενικές πληροφορίες	3
Χειρισμός	3
Οθόνη παιχνιδιού	3
Συνθήκες τέλους	3
Τεχνικές Πληροφορίες	4
Κλάσεις	4
Μεθόδοι.....	4
Λειτουργία Προγράμματος.....	4

Εργασία

Η εργασία υλοποιεί ένα παιχνίδι-λαβύρινθο 2 χαρακτήρων. Ο ένας χαρακτήρας ελέγχεται από τον χρήστη(κόκκινο χρώμα) , και ο δεύτερος από το πρόγραμμα(πράσινο χρώμα) . Στόχος κάθε παίκτη είναι να βρεθεί το πετράδι μέσα στον λαβύρινθο, το οποίο μετακινείται τυχαία και σε άτακτες χρονικές στιγμές.

Γενικές πληροφορίες

Χειρισμός

Ο παίκτης μετακινείται με τα βελάκια σε 4 κατευθύνσεις: πάνω, κάτω, αριστερά και δεξιά. Μπορούμε να βγούμε από το πρόγραμμα οποιαδήποτε στιγμή πατώντας το πλήκτρο 'Esc'.

Οθόνη παιχνιδιού

Στην οθόνη εμφανίζεται ο χάρτης του λαβύρινθου, οι 2 χαρακτήρες και το πετράδι.

Κάτω από το χάρτη εμφανίζονται οι εξής πληροφορίες:

- Εναπομείναντα βήματα έως την μετακίνηση του πετραδιού.
- Συντεταγμένες χαρακτήρων και πετραδιού (x,y).

Συνθήκες τέλους

Το παιχνίδι τελειώνει όταν ένας παίκτης βρει το πετράδι ή σταματάει όταν ο χρήστης πατήσει 'Esc'. Και στις δυο περιπτώσεις, εμφανίζεται μήνυμα 'GAME OVER' στο terminal του χρήστη.

Τεχνικές Πληροφορίες

Κλάσεις

- Main
- GameLogic
 - Μεταβλητές αντικειμένων:
 - Int PlayerX,playerY
 - Int PcX,PcY
 - Int RubyX,RubyY
- MakeMap

Μεθόδοι

GameLogic

- Int GetInfo(int)
- Void SetInfo(int,int)
- Void MovePlayer(int)
- Void MovePcCharacter(int *)
- Void MoveRuby(int *,int *)
- Int checkTiles(int *)
- Bool checkIfEnd()
- Void printPositions();

MakeMap

- Int createMap(string)
- Vector<int> initializeConditions()

Λειτουργία Προγράμματος

Τρέχουμε το πρόγραμμα με την εντολή `make run ARGS=<path>` όπου path είναι το relative path ενός txt αρχείου το οποίο περιέχει τον χάρτη. Π.χ. `make run ARGS=map.txt`

Αρχικά το πρόγραμμα θα δημιουργήσει τον χάρτη. Θα δημιουργήσει ένα αντικείμενο της κλάσης MakeMap και θα καλέσει τη μέθοδο createMap με argument το path που εισάγαμε τρέχοντας το πρόγραμμα. Στη συνέχεια αρχικοποιεί τις θέσεις των χαρακτήρων και του πετραδιού, με την μέθοδο initializeConditions. Η initializeConditions παράγει τα αναγκαία x και y μέσω της rand() και τα τοποθετεί σε ένα vector 6 θέσεων, το οποίο και γυρνάει στην main.

Πίσω στη main, το vector που παράχθηκε από την initializeConditions χρησιμοποιείται για να αρχικοποιήσει ένα αντικείμενο της κλάσης GameLogic. Η GameLogic είναι η βασική κλάση του προγράμματος και ελέγχει τις κινήσεις των παικτών και του πετραδιού.

Η βασική λειτουργία του προγράμματος βρίσκεται μέσα σε ένα while loop. Το loop σταματάει όταν η μεταβλητή run γίνει false ή όταν πατηθεί το 'Esc'.

Μετακίνηση πετραδιού:

Το πετράδι μετακινείται μέσω της moveRuby και έχει την εξής λογική. Μέσω 2 μεταβλητών step και StepGoal ελέγχεται η τυχαία χρονική και τοπική μετακίνηση. Η stepGoal αρχικοποιείται μέσω της rand() και παίρνει τιμές μεταξύ 50 και 150. Το εύρος τιμών επιλέχθηκε με βάση το μέγεθος των χαρτών(40x20). Για κάθε επανάληψη του while(δηλαδή κάθε μετακίνηση κατά ένα των παικτών), το step(βήμα) αυξάνεται κατά ένα. Μετά από StepGoal βήματα, η moveRuby δίνει μια νέα τιμή στη stepGoal και νέες τιμές στη συντεταγμένες του πετραδιού.

Μετακίνηση παίκτη:

Η κίνηση του παίκτη πραγματοποιείται μέσω της μεθόδου movePlayer. Ανάλογα το κουμπί που πατήθηκε αλλάζεται η ανάλογη συντεταγμένη του παίκτη κατά 1.

Η μέθοδος ξανατρέχει αναδρομικά σε περίπτωση που πατήθηκε ένα κουμπί μετακίνησης, αλλά η μετακίνηση είναι αδύνατη (λόγω τοίχου). Η λειτουργία αυτή ελέγχεται μέσω της εσωτερικής μεταβλητής moved.

Μετακίνηση υπολογιστή:

Η κίνηση του χαρακτήρα που ελέγχεται από τον υπολογιστή πραγματοποιείται μέσω της μεθόδου movePcCharacter.

Αρχικά, και μέσα στη movePcCharacter, καλείται η μέθοδος checkTiles. Η checkTiles δέχεται ως όρισμα την κατεύθυνση στην οποία 'διαλέγει' να μετακινηθεί το πετράδι και παίρνει τιμές 1=πάνω, 2=κάτω, 3=αριστερά, 4=δεξιά.

Η λογική έχει ως εξής:

- Δημιουργούμε ένα πίνακα 4 θέσεων, στον οποίο αποθηκεύονται οι ελεύθερες γειτονικές διευθύνσεις. Πρώτη θέση του πίνακα = πάνω, δεύτερη θέση = κάτω, τρίτη = αριστερά, τέταρτη = δεξιά.

- Ας υποθέσουμε ότι το πετράδι 'θέλει' να μετακινηθεί πάνω(1) και έρχεται από κάτω. Το πρόγραμμα ελέγχει τις διευθύνσεις πάνω, δεξιά και αριστερά. Αν μια κατεύθυνση είναι ελεύθερη, δηλαδή η θέση περιέχει '.', τότε η αντίστοιχη θέση στον πίνακα παίρνει την τιμή 1. Έστω μετά τον έλεγχο ο πίνακας είναι [1,0,1,0]. Δηλαδή μπορεί να γίνει μετακίνηση πάνω και αριστερά. Φυσικά μπορεί να μετακινηθεί και προς τα πίσω, δηλαδή προς τα κάτω.(Την περίπτωση αυτή θα την δούμε παρακάτω).
- Έχοντας το πίνακα με τις ελεύθερες διευθύνσεις, δημιουργείται τυχαία(rand()) μια τιμή μεταξύ 0 και 3. Η τιμή αυτή θεωρείται ως δείκτης σε μια θέση του πίνακα. Πχ δημιουργείται η τιμή 2. Ελέγχεται η θέση [2] του πίνακα. Αν η θέση [2] έχει την τιμή '1', τότε διαλέγεται αυτή ως νέα κατεύθυνση. Αν δεν είναι '1', τότε η τιμή-δείκτης αυξάνεται κατά μια μονάδα και ελέγχεται η αντίστοιχη θέση στον πίνακα, μέχρι να βρεθεί σωστή κατεύθυνση.
- Πριν κάνει την παραπάνω λειτουργία, ο πίνακας συγκρίνεται με έναν πίνακα[0,0,0,0]. Η σύγκριση γίνεται με σκοπό να καλυφθεί η περίπτωση όπου δεν υπάρχουν ελεύθερες κατευθύνσεις, και το πρόγραμμα πρέπει να κινηθεί αναγκαστικά προς τα πίσω. Έτσι για παράδειγμα, αν μετακινείται προς τα πάνω, και οι κατευθύνσεις πάνω, δεξιά και αριστερά δεν είναι ελεύθερες, τότε η νέα κατεύθυνση τίθεται προς τα κάτω.

Έχοντας βρει μια κατεύθυνση, η checkTiles επιστρέφει στην movePcCharacter η οποία πραγματοποιεί την μετακίνηση ,ανάλογα την κατεύθυνση, αλλάζοντας την αντίστοιχη συντεταγμένη του χαρακτήρα.

Πίσω στη main, αφού πραγματοποιηθεί η κίνηση του υπολογιστή, καλείται η checkIfEnd η οποία με μια απλή if ελέγχει αν οι συντεταγμένες του χρήστη, ή του υπολογιστή είναι ίδιες με τις συντεταγμένες του πετραδιού. Αν ισχύει μια από τις δύο συνθήκες, το πρόγραμμα βγαίνει από το βρόχο while.

Τέλος τυπώνει στην οθόνη τις συντεταγμένες των παικτών και του πετραδιού.

ΤΕΛΟΣ ΕΡΓΑΣΙΑΣ