

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

Άσκηση 2 “Προγραμματισμός Συστήματος” 2007-08 (Προθεσμία 27/11 – 18:00)

Γράψιμο ενός παιχνιδιού PacMan σε τερματικό.

Στόχος: Εξοικείωση με χρήση σημάτων (signals) και χρονομέτρων (timers).

Το παιχνίδι θα επιτρέπει στο χρήστη να ελέγχει τον Pacman χρησιμοποιώντας το πληκτρολόγιο. Κάθε μικρή χρονική περίοδος το παιχνίδι πρέπει να κινήσει τα ελεγχόμενα από τον υπολογιστή φαντάσματα κατά τρόπο ευφυή (δηλ. να το κάνετε να μοιάσει σαν να κυνηγάνε τον Pacman).

Η “γραφική απεικόνιση” θα γίνει χρησιμοποιώντας το παρακάτω κείμενο:

- Με ένα “C” για τον Pacman
- Με ένα “G” για κάθε ένα από τα 4 φαντάσματα
- Με ένα “.” για κάθε κανονικού σημείου που ο Pacman μπορεί να φάει
- Με ένα “P” για κάθε Χάπι Δύναμης, το οποίο όταν φάει ο Pacman του δίνει τη δύναμη να φάει τα φαντάσματα για μια μικρή χρονική περίοδο (5 δευτερόλεπτα).
- Με ένα “W” για κάθε τοίχο
- Με ένα “w” για το σπίτι των φαντασμάτων
- Με ένα κενό χαρακτήρα για κάθε σημείο που ο Pacman δεν παίρνει πόντους αν φάει

Στο μάθημα διδαχτήκαμε πώς μπορούμε να απεικονίσουμε χαρακτήρες και κείμενο σε συγκεκριμένα σημεία της οθόνης με τη χρήση της βιβλιοθήκης curses. Παρόμοιες συναρτήσεις θα χρειαστούν και σε αυτή την άσκηση. Επειδή στο παιχνίδι υπάρχουν 5 αντικείμενα που κινούνται (ο Pacman και τα 4 φαντάσματα), η απεικόνιση της κίνησης στην οθόνη είναι καλύτερο να γίνεται ως εξής:

- Η οθόνη ανανεώνεται περιοδικά. Πχ, αν θέλουμε να εξετάζουμε τι συμβαίνει στο παιχνίδι κάθε 40 ms, τότε ο παρακάτω κώδικας βοηθάει να επιτευχθεί αυτό (Μετά από την πρώτη κλήση στο setitimer (), το χρονόμετρο θα πυροδοτήσει κάθε 40.000 μικροδευτερόλεπτα, καλώντας οτιδήποτε handler ορίσουμε για το σήμα SIGALRM.):

```
/* for time measurements, delay between shape movements */
struct itimerval h;

/* define the first timer to set off after 40,000 microseconds. */
h.it_value.tv_sec = 0;
h.it_value.tv_usec = 40000;
/* define the interval between each two timer set-offs ('clicks') */
/* to be 40,000 micro seconds as well. */
h.it_interval.tv_sec = 0;
h.it_interval.tv_usec = 40000;

/* finally, start off the timer */
setitimer(ITIMER_REAL,&h,0);
```

- Την επόμενη φορά που θα χτυπήσει ο μετρητής, υπολογίζεται η νέα θέση του Pacman και των φαντασμάτων. Για αυτό το βήμα θα χρειαστείτε την κατάσταση του παιχνιδιού στην προηγούμενη ανανέωση της οθόνης. Χρησιμοποιήστε ένα διδιάστατο πίνακα για να αποθηκεύσετε την πληροφορία κάθε οθόνης.
- Όταν ανανεώνετε την οθόνη, συγκρίνετε τα στοιχεία της παλιάς και της νέας οθόνης και ανανεώστε ΜΟΝΟ τα σημεία που διαφέρουν.

Κίνηση χαρακτήρων

Το παιχνίδι έχει επίπεδα. Ο Pacman ξεκινάει με 3 ζωές. Ο Pacman έχει μία ορισμένη από τη γραμμή εντολών περίοδο κίνησης (ακέραιος αριθμός). Τα φαντάσματα έχουν μία (διαφορετική), επίσης δοσμένη από τη γραμμή εντολών, περίοδο κίνησης (ακέραιος αριθμός). Η περίοδος κίνησης ουσιαστικά ορίζει κάθε πόσες εκπυρσοκροτήσεις του μετρητή που αναφέραμε στην ανανέωση της οθόνης θα μετακινείται κατά ένα τετράγωνο (χαρακτήρα) της οθόνης το κάθε αντικείμενο (Pacman/φάντασμα). Πχ, περίοδος κίνησης 10 για τον Pacman σημαίνει ότι διασχίζει ένα χαρακτήρα κάθε $40ms \cdot 10 = 0.4$ δευτερόλεπτα.

Στο πρώτο επίπεδο του παιχνιδιού, ο Pacman είναι λογικό να κινείται πιο γρήγορα από τα φαντάσματα. Σε κάθε επίπεδο η περίοδος κίνησης κάθε φαντάσματος μειώνεται (άρα κινείται πιο γρήγορα) με βάση ένα τρίτο όρισμα που δίνεται στη γραμμή εντολών (ακέραιος αριθμός).

ΠΡΟΣΟΧΗ: Έτσι όπως ορίσαμε την περίοδο κίνησης, αύξηση της ταχύτητας σημαίνει μείωση των απαιτούμενων εκπυρσοκροτήσεων του μετρητή για την κίνηση του αντικειμένου. Πχ, αν η μεταβολή της περιόδου κίνησης ανά επίπεδο είναι 2, και η προηγούμενη περίοδος κίνησης των φαντασμάτων ήταν 12, τότε η περίοδος κίνησης στο επόμενο επίπεδο θα είναι $12 - 2 = 10$.

Σημειώστε ότι σε κάθε πλευρά του παιχνιδιού μπορεί να υπάρχει ένας ελλείπων τοίχος - αυτό διαμορφώνει τη σήραγγα που επιτρέπει Pac (καθώς επίσης και τα φαντάσματα) να μεταφέρονται από μια πλευρά της οθόνης στην άλλη σε μια στιγμή.

Ο Pacman αν συναντήσει τοίχο σταματάει. Τα φαντάσματα ΔΕΝ μπορούν να αντιστρέψουν την κίνηση τους. Έτσι, μπορούν να αλλάξουν κατεύθυνση ΜΟΝΟ σε διασταυρώσεις. 2 φαντάσματα ΔΕΝ μπορούν να βρίσκονται ταυτόχρονα στο ίδιο τετράγωνο. Σε περίπτωση που ένα φάντασμα Α πάει να κινηθεί σε τετράγωνο που υπάρχει άλλο φάντασμα Β, το φάντασμα Α θα πρέπει να θεωρηθεί ότι συνάντησε τοίχο και να αλλάξει πορεία.

Τα φαντάσματα πρέπει να κινούνται με κάποιον έξυπνο τρόπο προς τον Pacman. Πχ, αν ο Pacman είναι κάτω δεξιά ως προς το φάντασμα, υπάρχει λόγος να προτιμηθεί από το φάντασμα να κινηθεί προς τα πάνω (γενικώς υπάρχει, αλλά σπάνια)? Επίσης αν 2 φαντάσματα κυνηγούν τον Pacman, πώς μπορούν να τον παγιδεύσουν? Μία λύση είναι να επιλέγονται δύο δυνατές κατευθύνσεις σε κάθε διασταύρωση (πχ, στο παράδειγμά μας κάτω και δεξιά) και με τυχαίο τρόπο κάθε φάντασμα να επιλέγει μία από τις 2 (ή πχ με

πιθανότητα 2/3 να διαλέγει την καλύτερη δυνατή πορεία και με πιθανότητα 1/3 την δεύτερη καλύτερη επιλογή).

Τα φαντάσματα **αντιστρέφουν** την πορεία τους (χωρίς να περιμένουν να φτάσουν σε διασταύρωση) όταν ο Pacman φάει ένα Χάπι Δύναμης και προσπαθούν να απομακρυνθούν, μέχρι ένας μετρητής να σημάνει τη λήξη του Χαπιού (μετά από 4 δευτερόλεπτα). Κάθε φάντασμα που τρώγεται από τον Pacman εξαφανίζεται και εμφανίζεται μέσα στο σπίτι, σε μία κενή θέση του. Κάθε φάντασμα θα πρέπει να μένει στο σπίτι με τα φαντάσματα τουλάχιστον 2.4 δευτερόλεπτα (το κάθε φάντασμα έχει δικό του μετρητή, ανάλογα με το πότε το έφαγε ο Pacman). Το πώς θα εξέρχονται από το σπίτι είναι δικιά σας ιδέα. Θα μπορεί αυτόματα, μετά τα 2.4 δευτερόλεπτα, αν η θέση έξω από το σπιτάκι δεν καταλαμβάνεται από άλλο φάντασμα να βγαίνει ένα φάντασμα. Θα μπορούσατε να ορίσετε ένα τρόπο ώστε τα φαντάσματα να βρίσκουν μόνα τους πώς να βγουν. Ένα φάντασμα που έχει βγει από το σπιτάκι ΔΕΝ μπορεί να ξαναμπει μέσα παρά μόνο αν φαγωθεί ή αλλάξει η πίστα.

Ανάγνωση χαρακτήρων

Σκοπός είναι το πρόγραμμα να δέχεται εντολές από το πληκτρολόγιο για την κίνηση του Pacman. Προκειμένου να επιτραπεί στο πρόγραμμα να αντιμετωπίζει και την εισαγωγή χαρακτήρων και τους συναγερμούς (alarms) των timers, το πρόγραμμά μας θα χρησιμοποιήσει τη read() συνάρτηση, για την είσοδο του χρήστη, και **ΟΧΙ** η getch() συνάρτηση. Αυτό είναι επειδή η λήψη ενός σήματος θα κάνει την read() να επιστρέψει αμέσως (αυτό υποθέτει ότι ο χειριστής σημάτων για SIGALRM καθορίστηκε χρησιμοποιώντας το sigaction(), μην χρησιμοποιώντας τη signal()). Κατά συνέπεια, ο κύριος βρόχος θα αρχίσει με read() στον περιγραφητή αρχείων 0 (που χρησιμοποιείται για την τυποποιημένη εισαγωγή (standard input)), και έπειτα θα ακολουθεί έλεγχος για τον κώδικα επιστροφής. Εάν είναι -1, και το errno είναι ίσο με EINTR, κατόπιν πήραμε πιθανώς ένα SIGALRM, και πρέπει έτσι να καλέσουμε τα στοιχεία που κινούν τη λειτουργία. Διαφορετικά, ο χρήστης πίεσε πιθανώς ένα πλήκτρο, και πρέπει να ελέγξουμε ποιο. **Το πλήκτρο καταγράφεται αλλά λαμβάνεται υπόψιν την επόμενη φορά που ανανεώνεται η θέση του Pacman (όταν μεταβεί στην επόμενη θέση του, όχι νωρίτερα).** Επιπλέον των πλήκτρων κίνησης, το πλήκτρο 'Q' σημαίνει τον τερματισμό του προγράμματος, ενώ το πλήκτρο 'N' σημαίνει τη μετάβαση στο αμέσως επόμενο επίπεδο. **Κάθε πίστα να ξεκινάει όταν ο παίκτης πατήσει το γράμμα 'S' (όχι νωρίτερα).** Κάθε φορά που ένα φάντασμα τρώει τον Pacman, να παγώνει (να μην μετακινείται τίποτα δηλαδή) η οθόνη για 5 δευτερόλεπτα, ώστε να είναι εμφανές το γεγονός.

Τα σήματα SIGINT (CTRL-c) και SIGTSTP (CTRL-z) να καθαρίζουν την οθόνη και να τερματίζουν το πρόγραμμα. Θα χρειαστείτε να ορίσετε κατάλληλους χειριστές σημάτων.

Συνοπτικά

- Γράψτε έναν χειριστή σημάτων για SIGALRM που θέτει μια σημαία που δηλώνει ότι ένας συναγερμός παραλήφθηκε. Η μετακίνηση των αντικειμένων και η ανανέωση της οθόνης θα αντιμετωπίζονται εδώ.
- Γράψτε μια λειτουργία που κινεί Pac στην τρέχουσα κατεύθυνσή του, εμποδίζοντας τον εάν αναπηδά σε έναν τοίχο, που να τρώει ένα χάπι (και που να αυξάνει το αποτέλεσμα) εάν το νέο σημείο περιέχει ένα χάπι, που να τον θέτει σε Κατάσταση Δύναμης εάν τρώει ένα χάπι δύναμης (ή να τον επαναφέρει σε κανονική κατάσταση αν ήταν πριν σε Κατάσταση Δύναμης και πέρασε ο χρόνος δράσης του), που να πεθαίνει ο Pacman εάν πάτησε σε ένα φάντασμα (ή δολοφονία του φαντάσματος, εάν αυτό συμβαίνει ενώ ο Pacman είναι σε Κατάσταση Δύναμης).
- Γράψτε μια λειτουργία που κινεί ένα φάντασμα, που σκοτώνει Pac εάν πέσει πάνω στον Pacman (ή που σκοτώνει το φάντασμα, εάν ήταν σε Κατάσταση Δύναμης). Μια απλή λειτουργία θα προσπαθήσει ακριβώς να το φτάσει πιο κοντά σε Pac. Σημειώστε ότι ένα φάντασμα δεν μπορεί ποτέ να αντιστρέψει την κινούμενη κατεύθυνσή του. Μπορεί να επιλέξει να συνεχιστεί προς τα εμπρός, να κινηθεί αριστερά, ή να κινηθεί δεξιά. Έτσι γίνεται στο αρχικό παιχνίδι. **ΣΗΜΕΙΩΣΗ: Ο Pacman και το κάθε φάντασμα (και ΜΟΝΟ αυτοί οι χαρακτήρες) να απεικονίζονται σε μορφή standout για να φαίνονται πιο εύκολα στην οθόνη.**
- Γράψτε μια λειτουργία που εκτελεί όλες τις μετακινήσεις χαρακτήρων: κίνηση Pac και κίνηση των φαντασμάτων. Γράψτε μια λειτουργία που παίρνει δύο σειρές, μια περιγράφοντας το τρέχον σχεδιάγραμμα οθόνης, άλλο που περιγράφει το νέο (επιθυμητό) σχεδιάγραμμα οθόνης. Η λειτουργία θα ανανεώνει την οθόνη όπως περιγράψαμε νωρίτερα.
- Τέλος, η κύρια λειτουργία εγγράφεται τους χειριστές σημάτων, αρχίζει το χρονόμετρο (με το setitimer ()) και εκτελεί τον κύριο βρόχο της αναμονής την εισαγωγή χρηστών (ή τους συναγερμούς).
- Απεικονίστε κάπου στην οθόνη το Score του παίχτη, τις υπολειπόμενες ζωές του, το επίπεδο, την περίοδο κίνησης του Pacman, την αντίστοιχη περίοδο των Φαντασμάτων, και το χρόνο που μένει σε δευτερόλεπτα μέχρι να βγει ο παίκτης από την Κατάσταση Δύναμης. Επίσης, το πόσα στοιχεία απομένουν να φαγωθούν από τον Pacman για το τέλος της πίστας.

Score

10 πόντους για κάθε τελεία ‘.’

60 πόντους για κάθε Χάπι Δύναμης

300 πόντοι για το πρώτο φάντασμα που τρώει ο Pacman με ένα χάπι Δύναμης (αντίστοιχα, 600, 900, 3200 για το δεύτερο, τρίτο και τέταρτο φάντασμα με το ίδιο χάπι)

5000 πόντοι για τερματισμό επιπέδου (αν πατήσει ο χρήστης το ‘N’, ΔΕΝ υπάρχει αυτό το bonus)

Βαθμολογία

- Η βαθμολογία θα εξαρτηθεί από το πόσο ρεαλιστικό θα είναι το παιχνίδι (κυρίως η κίνηση των φαντασμάτων). Δεν περιμένω κάτι αντίστοιχο του πραγματικού παιχνιδιού, αλλά όχι να είναι πχ ο Pacman κάτω δεξιά και τα φαντάσματα πάνω αριστερά... Θα δείτε ότι δεν είναι πολύ δύσκολο, ιδίως για 2 άτομα.
- Μπορείτε να αλλάξετε τη συχνότητα του ρολογιού, αν πιστεύετε ότι έτσι θα πετύχετε πιο ρεαλιστική αναπαράσταση στην οθόνη. Αν το κάνετε αυτό θα πρέπει να το αναγράψετε στο README στην παράδοση της άσκησης.
- Μπορείτε να σκεφτείτε κάτι έξυπνο για την κίνηση των φαντασμάτων? Πχ, μπορούν να συνεργαστούν ώστε να εγκλωβίζουν πιο εύκολα τον Pacman?
- Πώς επιλέγουν την κίνησή τους τα φαντάσματα λαμβάνοντας υπόψιν τις εξόδους διαφυγής? Καταλαβαίνουν ότι ακόμα και αν είναι στην απέναντι πλευρά της οθόνης, μπορούν να πλησιάσουν τον Pacman μέσω των εξόδων διαφυγής?
- **Όλα όσα ανέφερα για χρήσεις συναρτήσεων είναι ΥΠΟΧΡΕΩΤΙΚΑ. Αυτό σημαίνει ότι όποιος ΔΕΝ χρησιμοποιήσει τις συναρτήσεις read, sigaction, ή δεν υλοποιήσει την κίνηση των φαντασμάτων και του Pacman με την περίοδο κίνησης, όπως ορίστηκε παραπάνω θα χάσει αρκετά μόρια.**
- Σε περίπτωση που οποιοδήποτε κομμάτι κώδικα ή τρόπος υλοποίησης συνάρτησης βγει στη λίστα ή στο forum ο διδάσκων μπορεί να αλλάξει τις απαιτήσεις της άσκησης οποιαδήποτε στιγμή, ή σε ακραία περίπτωση να ακυρώσει την άσκηση και να ορίσει νέα στη θέση της.
- **Εννοείται ότι διαφορετικές ομάδες ΔΕΝ μπορούν να έχουν την ίδια υλοποίηση οποιωνδήποτε συναρτήσεων. Κάτι τέτοιο θα θεωρηθεί αντιγραφή. Και δεν θα αποτελεί δικαιολογία αν κάποιος/κάποια έχει βγάλει το συγκεκριμένο κομμάτι κώδικα στο δίκτυο.**

Λογικά Βήματα

- Διαβάστε το αρχείο της πίστας και απεικονίστε το στην οθόνη με όλα τα απαραίτητα (score, ταχύτητα Pacman, ταχύτητα φαντασμάτων, ζώες κτλ)
- Ξεκινήστε από το πώς διαβάζονται οι επιθυμίες του χρήστη και ορίζεται ο χρόνος μεταξύ ανανεώσεων της οθόνης.
- Υλοποιήστε την κίνηση του Pacman στην πίστα χωρίς φαντάσματα.
- Προσθέστε τις λειτουργίες που σχετίζονται με τα φαντάσματα.

Δεληγιαννάκης Αντώνιος