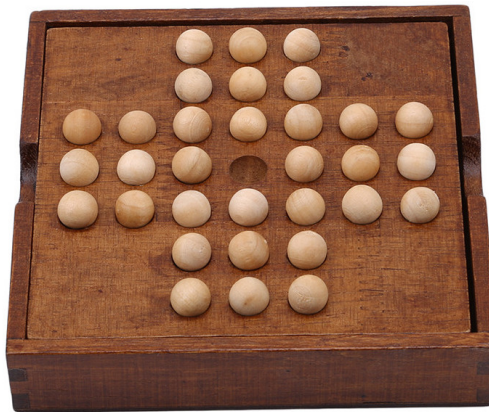


ΜΥΥ105: Εισαγωγή στον Προγραμματισμό
Χειμερινό Εξάμηνο 2021-22
Προγραμματιστική Εργασία 2021

Προθεσμία: Δευτέρα 20 Δεκεμβρίου 2021, ώρα 9μμ

Εισαγωγή - περιγραφή του παιχνιδιού

Το παιχνίδι “peg solitaire” ή απλά “solitaire” είναι ένα επιτραπέζιο παιχνίδι για έναν παίκτη, το οποίο παίζεται σε ένα ταμπλό με 33 υποδοχές (θέσεις) και 32 όμοια πιόνια (στην αρχή του παιχνιδιού), όπως δείχνει το Σχήμα 1. Στόχος του παίκτη είναι στο τέλος του παιχνιδιού να μείνει με όσο λιγότερα πιόνια γίνεται. Σε κάθε κίνηση ο παίκτης κινεί ένα πιόνι πάνω από ένα διπλανό του (οριζόντια ή κάθετα) ώστε να προσγειωθεί σε κάποια κενή θέση και αφαιρεί το πιόνι το οποίο υπερπήδησε. Το παιχνίδι τελειώνει μόλις εξαντληθούν οι δυνατές κινήσεις. Πλοηγηθείτε στη σελίδα <https://www.webgamesonline.com/peg-solitaire/> ώστε να παίξετε το παιχνίδι και να καταλάβετε τους κανόνες.



Σχήμα 1: Το παιχνίδι peg solitaire

Ζητούμενα εργασίας

Καλείστε να γράψετε ένα πρόγραμμα το οποίο σχεδιάζει και τυπώνει ένα ταμπλό solitaire και ζητάει από τον παίκτη να παίξει το παιχνίδι έως ότου δεν θα έχει άλλες κινήσεις. Μετά από κάθε κίνηση, τυπώνεται το ταμπλό και ζητείται από το χρήστη να εισάγει μία νέα κίνηση. Για παράδειγμα:

	1	2	3	4	5	6	7
A			1	1	1		
B			1	1	1		
C	1	1	1	1	1	1	1
D	1	1	1	0	1	1	1
E	1	1	1	1	1	1	1
F			1	1	1		
G			1	1	1		

Enter peg position followed by move (L, R, U, or D):

Τα σύμβολα “1” δηλώνουν θέσεις με πιόνια και τα “0” δηλώνουν κενές θέσεις. Η κίνηση του χρήστη είναι ένα αλφαριθμητικό τριών χαρακτήρων. Ο πρώτος χαρακτήρας πρέπει να είναι ένα γράμμα από το A έως το G και δηλώνει τη γραμμή στην οποία βρίσκεται το πιόνι που επιθυμεί να μετακινήσει ο χρήστης. Ο δεύτερος χαρακτήρας είναι ένα ψηφίο από το 1 έως το 7 και δηλώνει τη στήλη του πιονιού προς μετακίνηση. Τέλος, ο τρίτος χαρακτήρας είναι ένας από τους L, R, U, D, και δηλώνει την κατεύθυνση της μετακίνησης (L=left, R=right, U=up, D=down). Το πρόγραμμα μετακινεί το πιόνι με βάση την εντολή του χρήστη και ενημερώνει το ταμπλό. Για παράδειγμα, αν ο χρήστης δώσει στην είσοδο F4U, αυτό σημαίνει ότι θέλει να μετακινήσει το πιόνι στη θέση

F4 προς τα πάνω. Το πρόγραμμα μετακινεί το πιόνι με βάση την εντολή του χρήστη, θέτοντας 1 στην νέα θέση και 0 στην αρχική θέση του πιονιού και την θέση που του πιονιού που υπερπήδησε, και τυπώνει το ενημερωμένο ταμπλό. Δηλαδή, αν ο χρήστης δώσει στην είσοδο F4U, το πιόνι που ήταν στη θέση F4, θα μετακινηθεί στη θέση D4, οπότε: η θέση D4 θα μαρκαριστεί με “1” (που σημαίνει κατειλημμένη με πιόνι) και οι θέσεις E4 και F4 θα γίνουν “0”, όπως φαίνεται παρακάτω.

	1	2	3	4	5	6	7
A			1	1	1		
B			1	1	1		
C	1	1	1	1	1	1	1
D	1	1	1	1	1	1	1
E	1	1	1	0	1	1	1
F			1	0	1		
G			1	1	1		

Enter peg position followed by move (L, R, U, or D):

Το πρόγραμμα θα πρέπει να δίνει τη δυνατότητα στο χρήστη να εισάγει την κίνησή του και με πεζά γράμματα. Για παράδειγμα, η κίνηση f4u είναι επιτρεπτή και ισοδυναμεί με την F4U (Hint: χρησιμοποιήστε τη μέθοδο upper της Python για μετατροπή από πεζά σε κεφαλαία πριν από την επεξεργασία της κίνησης).

Αν ο χρήστης δώσει μία μη επιτρεπτή κίνηση, τότε το πρόγραμμα θα πρέπει να του ζητήσει να την ξαναδώσει. Για παράδειγμα, αν ο χρήστης γράψει C4X τότε το πρόγραμμα θα πρέπει να τυπώσει “Direction is not L or R or U or D!”, δηλαδή ότι η κατεύθυνση δεν είναι έγκυρη. Υπάρχουν διαφορετικοί λόγοι πίσω από μία μη έγκυρη κίνηση, όπως δείχνουν τα παρακάτω παραδείγματα. Έστω ότι το ταμπλό είναι όπως φαίνεται παραπάνω. Δείτε τα παρακάτω παραδείγματα μη έγκυρων κινήσεων.

Enter peg position followed by move (L, R, U, or D): A1R
Given peg position is out of board!

Εδώ, η θέση εκκίνησης πιονιού A1 δεν υπάρχει περίπτωση να έχει πιόνι γιατί είναι εκτός των θέσεων που μπορούν να έχουν πιόνια.

Enter peg position followed by move (L, R, U, or D): f4u
Given peg position does not have a peg!

Εδώ, η θέση εκκίνησης πιονιού F4 είναι κενή, άρα δεν μπορεί να μετακινηθεί πιόνι από τη θέση F4.

Enter peg position followed by move (L, R, U, or D): c4k
Direction is not L or R or U or D!

Εδώ, η κατεύθυνση της κίνησης δεν είναι μία εκ των L, R, U, D.

Enter peg position followed by move (L, R, U, or D): A4L
Moving peg will fall out of bounds!

Εδώ, αν μετακινήσουμε το πιόνι στη θέση A4 στα αριστερά πάνω από το πιόνι στη θέση A3, αυτό θα προσγειωθεί σε θέση εκτός του ταμπλό (δηλαδή στη θέση A2 η οποία δεν είναι έγκυρη).

Enter peg position followed by move (L, R, U, or D): E3R
No peg at next position to jump over!

Εδώ, ο χρήστης ζητεί το πιόνι στη θέση E3 να μετακινηθεί δεξιά, πηδώντας πάνω από το πιόνι στη θέση E4, αλλά η θέση E4 είναι κενή, άρα η κίνηση δεν είναι έγκυρη.

Enter peg position followed by move (L, R, U, or D): d1r
Landing position is occupied!

Εδώ, ο χρήστης ζητεί να μετακινηθεί το πιόνι στη θέση D1 δεξιά πηδώντας πάνω από το πιόνι στη θέση D2, αλλά η θέση προσγείωσης D3 δεν είναι κενή, οπότε η κίνηση δεν είναι έγκυρη.

Enter peg position followed by move (L, R, U, or D): hahaha
Something wrong with your input!

Εδώ, ο χρήστης δίνει μία κίνηση, η οποία δεν βγάζει νόημα.

Το πρόγραμμά σας θα πρέπει να ελέγχει αν η κίνηση του χρήστη είναι έγκυρη και αν δεν είναι δεν θα την εκτελεί, αλλά θα ζητάει ξανά από το χρήστη να εισάγει μία έγκυρη κίνηση. Μετά από κάθε έγκυρη κίνηση, το πρόγραμμα, αφού τυπώσει το ταμπλό που προκύπτει από την εκτέλεσή της, ελέγχει αν υπάρχουν έγκυρες κινήσεις που μπορούν να εκτελεστούν στη συνέχεια. Αν ναι, ζητάει ξανά από το χρήστη να εισάγει την επόμενη του κίνηση, αλλιώς τερματίζει και τυπώνει τον αριθμό των εναπομείναντων πιονιών στο ταμπλό. Όσο μικρότερος είναι αυτός ο αριθμός τόσο καλύτερα έπαιξε ο παίκτης. (Στόχος του παίκτη είναι στο τέλος να μείνει με ένα πιόνι μόνο.) Για παράδειγμα:

	1	2	3	4	5	6	7
A		0	0	0			
B		0	0	0			
C	0	0	0	0	0	0	0
D	0	0	0	1	0	0	0
E	0	0	1	1	1	0	0
F		0	0	0			
G		0	0	0			

Enter peg position followed by move (L, R, U, or D): d4d

	1	2	3	4	5	6	7
A		0	0	0			
B		0	0	0			
C	0	0	0	0	0	0	0
D	0	0	0	0	0	0	0
E	0	0	1	0	1	0	0
F		0	1	0			
G		0	0	0			

No more moves. The number of remaining pegs is 3

Συνοπτικά το ζητούμενο πρόγραμμα περιγράφεται από τον παρακάτω ψευδοκώδικα:

1. αρχικοποίησε το ταμπλό και τύπωσέ το
2. για όσο υπάρχουν έγκυρες κινήσεις στο ταμπλό:
 - ζήτη από το χρήστη να εισάγει την επόμενη κίνηση, μέχρι η δοθείσα κίνηση να είναι έγκυρη
 - εκτέλεσε την κίνηση που εισήγαγε ο χρήστης
 - τύπωσε το ενημερωμένο ταμπλό
3. τύπωσε τον αριθμό των πιονιών που έμειναν και τερμάτισε

Υποδείξεις. Μπορείτε να κάνετε χρήση δισδιάστατου πίνακα (λίστα από λίστες) για να αποθηκεύσετε, να ενημερώνετε και να ελέγχετε τις θέσεις των πιονιών. Ενδείκνυται να γράψετε συναρτήσεις για την εκτύπωση του ταμπλό, για τον έλεγχο και την εκτέλεση μιας κίνησης, για τον έλεγχο ύπαρξης μιας έγκυρης κίνησης, κλπ.

ΟΔΗΓΙΕΣ ΓΙΑ ΤΗΝ ΥΠΟΒΟΛΗ ΤΗΣ ΕΡΓΑΣΙΑΣ

1. Σώστε το πρόγραμμά σας στο αρχείο `solitaire.py` και υποβάλετε το αρχείο κάνοντας χρήση `turnin`, όπως περιγράφεται στην ιστοσελίδα του μαθήματος. Η υποβολή πρέπει να γίνει *από το δικό σας λογαριασμό* και όχι από λογαριασμούς συμφοιτητών σας. Προσοχή: Δεν μπορείτε να υποβάλετε εργασία αν δεν έχετε λογαριασμό στα εργαστήρια. Εκπρόθεσμες υποβολές ή υποβολές μέσω email δεν θα γίνονται δεκτές.

2. Προσοχή: Το πρόγραμμά σας δεν πρέπει να έχει καθόλου ελληνικούς χαρακτήρες, γιατί αλλιώς μπορεί να αποτύχει το turnin.
3. Στην πρώτη γραμμή του αρχείου σας, *σε σχόλια*, θα πρέπει να αναγράφονται το ονοματεπώνυμο και ο ΑΜ σας *με λατινικούς χαρακτήρες*. Προγράμματα που δεν φέρουν όνομα και ΑΜ δεν θα βαθμολογηθούν.
4. Η εργασία σας θα πρέπει να ακολουθεί πιστά τις οδηγίες. Προγράμματα που παίζουν ένα παρόμοιο, αλλά όχι το ίδιο παιχνίδι δεν θα βαθμολογηθούν. Επίσης, δεν θα γίνονται δεκτά προγράμματα στα οποία η επικοινωνία με το χρήστη και η παρουσίαση είναι διαφορετική σε σχέση με την περιγραφή.
5. Εργασίες που έχουν αντιγραφεί τμηματικά ή εξ ολοκλήρου από τον παγκόσμιο ιστό ή από εργασίες συμφοιτητών σας θα μηδενίζονται. Προσοχή: Θα χρησιμοποιηθεί ειδικό λογισμικό το οποίο εντοπίζει αντιγραφές μεταξύ προγραμμάτων!
6. Ενδέχεται κάποιοι από τους φοιτητές να εξεταστούν προφορικά στο περιεχόμενο της εργασίας τους.