

ΕΡΓΑΣΙΑ 1 SUDOKU

ΝΙΚΟΛΑ ΣΙΑΚΑΣ 171011

ΣΥΝΑΡΤΗΣΕΙΣ:

`printTable`: δέχεται σαν όρισμα τον 2d πίνακα και τον εμφανίζει. Η πρώτη επανάληψη εμφανίζει τους αριθμούς των στηλών. Η δεύτερη επανάληψη εμφανίζει τους αριθμούς των γραμμών, τις γραμμές και τις στήλες και μέσα στα κουτιά που σχεδίασε ελέγχει αν πρέπει να βάλει αριθμό η `space`.

`findPosition`: δέχεται σαν ορίσματα τον 2d πίνακα, δυο pointers και έναν ακέραιο. Ο ακέραιος δηλώνει πόσα κελιά που έχουν τον χαρακτήρα 0 να προσπεράσει και επιστρέφει 0 αν δεν βρει κενό κελί αφού έχει προσπεράσει τα `s` κενά κελιά αλλιώς επιστρέφει 1 και τις συντεταγμένες του κελιού.

`missingNums`: δέχεται σαν ορίσματα τον 2d πίνακα, έναν πίνακα τύπου `char` και τις συντεταγμένες του κελιού που θέλουμε να βρούμε τις τιμές που μπορεί να πάρει. Δημιουργεί έναν πίνακα (`values`) στον οποίο τοποθετεί τις τιμές που βρίσκονται στην ίδια σειρά, στήλη και στο ίδιο 3*3 κουτί. Μετά με ένα `for` ελέγχει τις τιμές από το 1 στο 9 αν βρίσκονται στον πίνακα `values`, όποιος αριθμός δεν βρίσκεται μέσα στον πίνακα `values` τον τοποθετεί στον `missingValues` που πείρε σαν όρισμα και αν έβαλε μόνο έναν αριθμό στον `missingValues`, δηλαδή αν το συγκεκριμένο κελί μπορεί να πάρει μόνο μια τιμή, επιστρέφει 1 αλλιώς επιστρέφει 0.

`validNum`: δέχεται σαν ορίσματα τον 2d πίνακα, έναν ακέραιο (`num`) και τις συντεταγμένες ενός κελιού και επιστρέφει 1 αν ο `num` μπορεί να μπει σε αυτο το κελί αλλιώς επιστρέφει 0.

`findValue`: δέχεται σαν ορίσματα τον 2d πίνακα και τον αριθμό των κενών κελιών του παζλ (`m`). Δημιουργεί έναν πίνακα (`missingValues`) και κάνει μια επανάληψη, για `i` από το 0 μέχρι `m`, όπου `i` ο αριθμός των κενών κελιών που πρέπει να προσπεράσει η `findPosition`. Η επανάληψη θα τελειώσει όταν η `missingNums` επιστρέψει 1 δηλαδή όταν για ένα κενό κελί, του οποίου τις συντεταγμένες επέστρεψε η `findPosition`, βρει μόνο έναν αριθμό που μπορεί να μπει σε αυτή την θέση. Σε αυτήν την περίπτωση βάζει αυτόν τον αριθμό στο συγκεκριμένο κελί του πίνακα και επιστρέφει 1. Άμα δεν βρει κενό κελί που να παίρνει μόνο έναν αριθμό επιστρέφει 0.

`findValueRec`: δέχεται σαν ορίσματα τον 2d πίνακα και έναν πίνακα τύπου `struct rec` (`tel`). Βρίσκει το πρώτο κενό κελί και με μια επανάληψη, για `i` από 1 μέχρι 10, ελέγχει αν το `i` μπορεί να μπει σε αυτό το κελί, αν γίνεται να μπει σε αυτή την θέση το βάζει και καλεί αναδρομικά τον εαυτό του. Όταν σε κάποιο κενό κελί δεν μπορεί να μπει κανένας αριθμός από το 1 στο 9 τότε επιστρέφει 0 στην σημαία `flag` όπου κλήθηκε και για το προηγούμενο κελί αυξάνεται το `i` και δοκιμάζει να δει αν μπορεί να μπει κάποιος άλλος αριθμός στο κελί και να βρει κάποιον τότε ξανά καλεί τον εαυτό του, αν όχι τότε επιστρέφει στον προηγούμενο κελί και κάνει το ίδιο μέχρι να λύσει το παζλ. Όταν το λύσει πριν επιστρέφει 1 σε όλες τις κλήσεις και να τελειώσει η συνάρτηση τοποθετεί στο `tel` τον αριθμό της γραμμής, της στήλης και την τιμή του κελιού του οποίου έχει βρει σε αυτήν την αναδρομική κλήση και ξανά βάζει στον πίνακα 0 ώστε να τα ξανά τοποθετηθεί μετά στην `main` και να τα εμφανίσει ένα-ένα.

`main`:

`arx`: ένας πίνακας τύπου `struct rec` ο οποίος περιέχει τις θέσεις και τις τιμές του αρχικού πίνακα που διαβάσαμε από το αρχείο.

tel: ένας πίνακας τύπου struct rec ο οποίος χρησιμοποιείτε άμα χρειαστεί η αναδρομή, παίρνει τις συντεταγμένες και τις τιμές του τελικού πίνακα με σκοπό να τα εμφανίσει ένα-ένα.

pin: ο 2d πίνακας τον οποίο πρέπει να γεμίσει το πρόγραμμα. Ο pin έχει αρχικοποιηθεί στον χαρακτήρα 0 μετά από τον πίνακα arx παίρνει τις αρχικές τιμές στις οποίες προσθέτει 48 ώστε με το %c να εμφανίσουμε είτε τον χαρακτήρα είτε το space.

k: στην αρχή το k περιέχει τον αριθμό των διαβασμένων αριθμών από το αρχείο και μετά το περιέχει τον αριθμό των κελιών που λείπουν.

Η main καλεί την findValue με ένα while, όσο η findValue επιστρέφει 1, δηλαδή βρίσκει αριθμό, αφαιρεί 1 από το k και εμφανίζει την καινούργια κατάσταση του πίνακα. Όταν η findValue επιστρέψει 0 ελέγχει να δει αν το k μηδενίστηκε, στην περίπτωση που μηδενίστηκε τελιώνει το πρόγραμμα, αλλιώς δεσμεύει μνήμη για τον πίνακα tel και καλεί την αναδρομική συνάρτηση findValueRec και όταν τελειώσει η συνάρτηση με ένα for τοποθετεί τις τιμές που βρήκε η συνάρτηση στον πίνακα, κάθε φορά που τοποθετεί έναν αριθμό στον πίνακα εμφανίζει την καινούργια κατάσταση του πίνακα.

ERROR CODES:

- 1: λάθος αριθμός παραμέτρων στην κλίση του προγράμματος
- 2: το πρόγραμμα δεν μπόρεσε να ανοίξει το αρχείο με το puzzle
- 3: δεν διάβασε σωστά το puzzle