

ΗΥ-100: Εισαγωγή στην Επιστήμη Υπολογιστών

3η σειρά ασκήσεων

Οδηγίες:

Για τη μεταγλώττιση των προγραμμάτων που ζητούνται θα πρέπει να χρησιμοποιήσετε το gcc

με τις παρακάτω παραμέτρους:

```
gcc -Wall -Werror
```

Για ευκολία, στα μηχανήματα του εργαστηρίου έχει δημιουργηθεί το alias gcc100 που καλεί τον gcc με αυτές τις παραμέτρους.

ΠΡΟΣΟΧΗ: Οι ασκήσεις θα βαθμολογηθούν με αυτόματο τρόπο, οπότε θα πρέπει να υπακούν πιστά την εκφώνηση (ακόμη και τα ονόματα των αρχείων). Για να δοκιμάσετε την άσκηση σας χρησιμοποιήστε τα test files που βρίσκονται στον gitlab repo της άσκησης hw2-tests εκτελώντας την εντολή:

```
sh test.sh ./hw3 tests
```

όπου hw3 είναι το όνομα του προγράμματος σας και tests ο φάκελος με τα test. Κάθε test αποτελείται από ένα αρχείο με κατάληξη .in, που περιέχει την είσοδο για το πρόγραμμα σας, και το αντίστοιχο .out αρχείο που περιέχει την αναμενόμενη έξοδο του προγράμματος.

Για κάθε test που περνάει το πρόγραμμά σας θα τυπώνεται **PASS**, αλλιώς **FAIL**.

Μαγικό Τετράγωνο

Μαγικό τετράγωνο είναι ένα $N \times N$ τετράγωνο το οποίο έχει τα εξής χαρακτηριστικά:

- Περιέχει όλους τους αριθμούς από 1 έως και N^2 .
- Το άθροισμα κάθε γραμμής, κάθε στήλης και κάθε μιας από τις δύο διαγωνίους είναι ίδιο και ίσο με $N \times (N^2 + 1)/2$.

Ένα παράδειγμα 3×3 μαγικού τετραγώνου είναι:

2	7	6	→15
9	5	1	→15
4	3	8	→15
↙15	↓15	↓15	↓15
			↘15

Σκοπός της άσκησης είναι να γράψετε ένα πρόγραμμα το οποίο θα μπορεί να ελέγχει αν ένας $N \times N$ πίνακας είναι μαγικό τετράγωνο, και επίσης να μπορεί να παίρνει σαν όρισμα έναν ακέραιο αριθμό N και να παράγει ένα $N \times N$ μαγικό τετράγωνο. Μπορείτε να υποθέσετε ότι $N < 100$, το οποίο σημαίνει ότι στην αρχή του προγράμματός σας θα πρέπει να ορίσετε ότι:

```
#define MAXN 100
```

Το πρόγραμμα θα εκτελεί δύο διαφορετικές λειτουργίες: τον έλεγχο και τη δημιουργία ενός μαγικού τετραγώνου. Για να ξεχωρίσει ποιιά απο τις λειτουργίες θα εκτελέσει, το πρόγραμμα θα δέχεται ένα επιπλέον command line argument το οποίο θα είναι είτε -create είτε -check.

1. Λειτουργία 1: Έλεγχος

Για να ελέγξουμε εάν ένα τετράγωνο είναι μαγικό θα πρέπει να εκτελούμε το πρόγραμμα hw3 με την παράμετρο γραμμής εντολών -check, ως εξής:

```
./hw3 -check
```

Το πρόγραμμα θα πρέπει να διαβάζει ένα πίνακα από το πληκτρολόγιο (χρησιμοποιώντας τη scanf). Ο πρώτος ακέραιος που διαβάζει είναι το N (η διάσταση του πίνακα). Στη συνέχεια, το πρόγραμμα πρέπει να διαβάζει κάθε ένα από τα στοιχεία του πίνακα (συνολικά N^2).

Ενδεικτικό παράδειγμα εισόδου:

```
3
2   7   6
9   5   1
4   3   8
```

Output

Το πρόγραμμά σας, αφού διαβάσει τον πίνακα, θα πρέπει να εξετάζει όλες τις γραμμές, στήλες, και τις δύο διαγωνίους, καθώς και το εάν τα στοιχεία του πίνακα είναι μοναδικά (δηλαδή περιέχει ακριβώς μία φορά τους αριθμούς 1 έως και N^2).

Για κάθε ένα από τα παραπάνω θα πρέπει να τυπώνει το είδος του ελέγχου (ROW, COLUMN, DIAG1, DIAG2, UNIQUE) και να τυπώνει YES ή NO αναλόγως αν ο έλεγχος πέτυχε

ή απέτυχε. Στο τέλος, θα αναφέρει αν το τετράγωνο είναι όντως μαγικό ή όχι, τυπώνοντας MAGIC YES ή MAGIC NO αντίστοιχα. Για παράδειγμα αν δώσουμε την παραπάνω είσοδο, η έξοδος θα πρέπει να είναι:

```
ROW 1 YES
ROW 2 YES
ROW 3 YES
COLUMN 1 YES
COLUMN 2 YES
COLUMN 3 YES
DIAG1 YES
DIAG2 YES
UNIQUE YES
MAGIC YES
```

Όπου DIAG1 είναι η διαγώνιος που αρχίζει άνω αριστερά και DIAG2 η διαγώνιος που αρχίζει πάνω δεξιά. Αντίθετα, αν δώσουμε σαν είσοδο:

```
3
2  7  1
9  5  1
4  3  8
```

η έξοδος θα πρέπει να είναι:

```
ROW 1 NO
ROW 2 YES
ROW 3 YES
COLUMN 1 YES
COLUMN 2 YES
COLUMN 3 NO
DIAG1 YES
```

DIAG2 NO

UNIQUE NO

MAGIC NO

Μπορείτε να υποθέσετε ότι οι αριθμοί που δίνονται στο πρόγραμμά σας θα είναι πάντα από 1 έως και N^2 .

1.2 Λειτουργία 2: Δημιουργία

Για να δημιουργήσουμε ένα μαγικό τετράγωνο θα πρέπει να εκτελούμε το πρόγραμμα hw3 με την παράμετρο γραμμής εντολών -create και στη συνέχεια έναν αριθμό N που είναι η διάσταση του πίνακα. Για παράδειγμα, για τη δημιουργία ενός μαγικού τετραγώνου διαστάσεων 9×9 , θα πρέπει το πρόγραμμα να εκτελεστεί ως εξής:

```
./hw3 -create 9
```

Αλγόριθμος

Μπορείτε να χρησιμοποιήσετε τον αλγόριθμο που περιγράφει ο παρακάτω ψευδοκώδικας για να υλοποιήσετε τη λειτουργία -create:

```
row = n-1
column = n/2
For k = 1 ... n*n
    Place k at [row][column].
    Increment row and column.
    If the row or column is n, replace it with 0.
    If the element at [row][column] is already filled
        Set row and column to their previous values.
    Decrement row.
```

Ο παραπάνω αλγόριθμος δουλεύει μόνο εάν το N είναι περιττός. Δε θα σας ζητηθεί να κάνετε create με N άρτιο.

Output

Σαν έξοδο το πρόγραμμα θα πρέπει να τυπώνει αρχικά τη διάσταση του πίνακα N, και στη συνέχεια όλα τα στοιχεία του πίνακα που θα δημιουργήσει, στοιχισμένα. π.χ.:

```
./hw3 -create 3
```

3

4

9

2

3

5

7

Υποδείξεις

Μπορείτε να χρησιμοποιήσετε ανακατεύθυνση για να τρέξετε το πρόγραμμα σας με μια συγκεκριμένη είσοδο:

```
./hw3 -check < tests/testX.in
```

Μπορείτε να δημιουργήσετε ένα τετράγωνο και να το αποθηκεύσετε σε ένα αρχείο test.out, για να το χρησιμοποιήσετε ως είσοδο αργότερα, καλώντας:

```
./hw3 -create 5 > test.out
```

Για να καλέσετε το hw3 με το test.out σαν είσοδο:

```
./hw3 -check < test.out
```

Χρήσιμες συναρτήσεις: scanf, atoi, strcmp.