Μάθημα 89

Δομές δεδομένων

Μονοδιάστατος σε δισδιάστατο

Εξετάσεις 2019 (επαναληπτικές) / Θέμα Β1

Ένας πίνακας λέγεται αραιός (sparse) αν ένα μεγάλο ποσοστό των στοιχείων του έχουν μηδενική τιμή.
Ένας δισδιάστατος αραιός πίνακας μπορεί να αναπαρασταθεί από έναν μονοδιάστατο όπου κάθε μη μηδενικό στοιχείο του δισδιάστατου αντιπροσωπεύεται στον μονοδιάστατο από μία τριάδα στοιχείων, δηλαδή <γραμμή, στήλη, τιμή>.
Για παράδειγμα, ο διπλανός πίνακας Α[4,5] που θέλουμε να τον διαχειριστούμε ως αραιό αντιπροσωπεύεται από τον μονοδιάστατο Β[15].

A	1	2	3	4	5
1	0	7	0	0	0
2	1	2	0	0	-3
3	0	0	4	0	0
4	0	0	0	0	0

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
B	1	2	**7**	2	1	**1**	2	2	**2**	2	5	**-3**	3	3	**4**

Η αντίστροφη διαδικασία είναι από τον μονοδιάστατο πίνακα να παραχθεί ένας ισοδύναμος αραιός δισδιάστατος.
Έστω ένας πίνακας Μ[18] που αναπαριστά 6 μη μηδενικά στοιχεία.
Δίνεται ο διπλανός αλγόριθμος, ο οποίος από τον μονοδιάστατο Μ[18] δημιουργεί τον αραιό δισδιάστατο Δ[10,20].
Να συμπληρώσετε τα κενά, ώστε να λειτουργήσει σωστά ο αλγόριθμος.

Αλγόριθμος αντίστροφος
  Δεδομένα // Μ //
  Για i **από** 1 **μέχρι** 20
    Για j **από** 1 **μέχρι** 10
      Δ [..j.. , ...i..] ← 0
    Τέλος_επανάληψης
  Τέλος_επανάληψης
  Για i **από** 1 **μέχρι** 18 **με_βήμα** ...3...
    α ← Μ[i]
    β ← Μ[i + ...1...]
    γ ← Μ[i + ...2...]
    Δ [α, β] ← γ
  Τέλος_επανάληψης
  Αποτελέσματα // Δ //
Τέλος αντίστροφος

Μετατροπή μετρητή σε συντεταγμένες

Αλγόριθμος που από τα αριθμημένα κελιά k = 1, 2, 3, . . . του παρακάτω πίνακα, δημιουργεί τις συντεταγμένες του i, j.

	1	2	3	4	5
1	(1 – 1) div 5 + 1 = 1 1 (1 – 1) mod 5 + 1 = 1	(2 – 1) div 5 + 1 = 1 2 (2 – 1) mod 5 + 1 = 2	(3 – 1) div 5 + 1 = 1 3 (3 – 1) mod 5 + 1 = 3	(4 – 1) div 5 + 1 = 1 4 (4 – 1) mod 5 + 1 = 4	(5 – 1) div 5 + 1 = 1 5 (5 – 1) mod 5 + 1 = 5
2	(6 – 1) div 5 + 1 = 2 6 (6 – 1) mod 5 + 1 = 1	(7 – 1) div 5 + 1 = 2 7 (7 – 1) mod 5 + 1 = 2	(8 – 1) div 5 + 1 = 2 8 (8 – 1) mod 5 + 1 = 3	(9 – 1) div 5 + 1 = 2 9 (9 – 1) mod 5 + 1 = 4	(10 – 1) div 5 + 1 = 2 10 (10 – 1) mod 5 + 1 = 5
3	(11 – 1) div 5 + 1 = 3 11 (11 – 1) mod 5 + 1 = 1	(12 – 1) div 5 + 1 = 3 12 (12 – 1) mod 5 + 1 = 2	(13 – 1) div 5 + 1 = 3 13 (13 – 1) mod 5 + 1 = 3	(14 – 1) div 5 + 1 = 3 14 (14 – 1) mod 5 + 1 = 4	(15 – 1) div 5 + 1 = 3 15 (15 – 1) mod 5 + 1 = 5
4	(16 – 1) div 5 + 1 = 4 16 (16 – 1) mod 5 + 1 = 1	(17 – 1) div 5 + 1 = 4 17 (17 – 1) mod 5 + 1 = 2	(18 – 1) div 5 + 1 = 4 18 (18 – 1) mod 5 + 1 = 3	(19 – 1) div 5 + 1 = 4 19 (19 – 1) mod 5 + 1 = 4	(20 – 1) div 5 + 1 = 4 20 (20 – 1) mod 5 + 1 = 5

i = (k – 1) div 5 + 1 **j = (k – 1) mod 5 + 1**

```
Αλγόριθμος μάθημα_89_k_σε_i_j
   Για k από 1 μέχρι 20
      i ← (k - 1) div 5 + 1
      j ← (k - 1) mod 5 + 1
      Εμφάνισε i, ",", j
   Τέλος_επανάληψης
Τέλος μάθημα_89_k_σε_i_j
```

Μετατροπή μονοδιάστατου σε δισδιάστατο / k συναρτήσει των i, j

Αλγόριθμος που διαβάζει τα στοιχεία ενός πίνακα ακεραίων A[6] και στη συνέχεια τα μεταφέρει στον πίνακα B[2, 3] , ανά γραμμή και τέλος εμφανίζει τα στοιχεία του B.

A

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

B

	1	2	3
1	3	5	7
2	8	4	6

```
Αλγόριθμος μάθημα_89_1D_ΣΕ_2D_1B
   Για k από 1 μέχρι 6
      Διάβασε A[k]
   Τέλος_επανάληψης
   Για k από 1 μέχρι 6
      i ← (k - 1) div 3 + 1
      j ← (k - 1) mod 3 + 1
      B[i, j] ← A[k]
   Τέλος_επανάληψης
   Για i από 1 μέχρι 2
      Εμφάνισε B[i, 1], B[i, 2], B[i, 3]
   Τέλος_επανάληψης
Τέλος μάθημα_89_1D_ΣΕ_2D_1B
```

Πρόβλημα 85 / σελίδα 204

Έστω ένας μονοδιάστατος πίνακας A[30]. Να δοθεί αλγόριθμος ο οποίος θα διαβάζει και θα ελέγχει για την καταλληλότητα τον αριθμό των γραμμών και τον αριθμό των στηλών ενός δισδιάστατου πίνακα B και θα αντιγράφει τον A στον B γραμμή-γραμμή.

```
Αλγόριθμος pro_4_85_A
   Για i από 1 μέχρι 30
      Διάβασε A[i]
   Τέλος_επανάληψης
   Αρχή_επανάληψης
      Διάβασε M, N
   Μέχρις_ότου M * N = 30
   k ← 1
   Για i από 1 μέχρι M
      Για j από 1 μέχρι N
         B[i, j] ← A[k]
         k ← k + 1
      Τέλος_επανάληψης
   Τέλος_επανάληψης
   Για i από 1 μέχρι M
      Για j από 1 μέχρι N
         Εμφάνισε B[i, j]
      Τέλος_επανάληψης
   Τέλος_επανάληψης
Τέλος pro_4_85_A
```

```
Αλγόριθμος pro_4_85_B
   Για k από 1 μέχρι 30
      Διάβασε A[k]
   Τέλος_επανάληψης
   Αρχή_επανάληψης
      Διάβασε M, N
   Μέχρις_ότου M * N = 30
   Για k από 1 μέχρι 30
      i ← (k - 1) div N + 1
      j ← (k - 1) mod N + 1
      B[i, j] ← A[k]
   Τέλος_επανάληψης
   Για i από 1 μέχρι M
      Για j από 1 μέχρι N
         Εμφάνισε B[i, j]
      Τέλος_επανάληψης
   Τέλος_επανάληψης
Τέλος pro_4_85_B
```

Πρόβλημα 89 / σελίδα 204 / Απευθείας (In situ) ταξινόμηση δισδιάστατου

Να δοθεί αλγόριθμος ο οποίος διαβάζει και καταχωρεί αριθμούς σε έναν δισδιάστατο πίνακα 5x6.
Στη συνέχεια εμφανίζει όλους τους αριθμούς του πίνακα ταξινομημένους σε αύξουσα σειρά.

```
Αλγόριθμος lesson_89_2D_SORT_ALL
   Για i από 1 μέχρι 4
      Για j από 1 μέχρι 5
         Διάβασε A[i, j]
      Τέλος_επανάληψης
   Τέλος_επανάληψης
   Για i από 2 μέχρι 20
      Για j από 20 μέχρι i με_βήμα -1
         i1 ← (j - 1) div 5 + 1
         j1 ← (j - 1) mod 5 + 1
         i2 ← (j - 2) div 5 + 1
         j2 ← (j - 2) mod 5 + 1
         Αν A[i2, j2] > A[i1, j1] τότε
            temp ← A[i2, j2]
            A[i2, j2] ← A[i1, j1]
            A[i1, j1] ← temp
         Τέλος_αν
      Τέλος_επανάληψης
   Τέλος_επανάληψης
   Για i από 1 μέχρι 4
      Για j από 1 μέχρι 5
         Εμφάνισε A[i, j]
      Τέλος_επανάληψης
   Τέλος_επανάληψης
Τέλος lesson_89_2D_SORT_ALL
```

A	1	2	3	4	5
1	13	34	22	41	21
2	24	12	43	45	31
3	14	33	42	11	25
4	23	35	15	44	32

A	1	2	3	4	5
1	11	12	13	14	15
2	21	22	23	24	25
3	31	32	33	34	35
4	41	42	43	44	45

Πρόβλημα 127 / σελίδα 218

(2011_Δ) Στην αρχή της ποδοσφαιρικής περιόδου οι 22 παίκτες μιας ομάδας, οι οποίοι αριθμούνται από 1 έως 22, ψηφίζουν για τους 3 αρχηγούς που θα τους εκπροσωπούν.
Κάθε παίκτης μπορεί να ψηφίσει όσους συμπαίκτες του θέλει, ακόμα και τον εαυτό του.
Τα αποτελέσματα της ψηφοφορίας καταχωρίζονται σε έναν πίνακα ΨΗΦΟΣ με 22 γραμμές και 22 στήλες, έτσι ώστε το στοιχείο ΨΗΦΟΣ[i, j] να έχει την τιμή 1, όταν ο παίκτης με αριθμό i έχει ψηφίσει τον παίκτη με αριθμό j, και τιμή 0 στην αντίθετη περίπτωση.
Να γράψετε αλγόριθμο ο οποίος:

Δ1. Να διαβάζει τα στοιχεία του πίνακα ΨΗΦΟΣ και να ελέγχει την ορθότητά τους με αποδεκτές τιμές 0 ή 1.

Μονάδες 4

Δ2. Να εμφανίζει το πλήθος των παικτών που δεν ψήφισαν κανέναν.

Μονάδες 4

Δ3. Να εμφανίζει το πλήθος των παικτών που ψήφισαν τον εαυτό τους.

Μονάδες 4

Δ4. Να βρίσκει τους 3 παίκτες που έλαβαν τις περισσότερες ψήφους και να εμφανίζει τους αριθμούς τους και τις ψήφους που έλαβαν. Θεωρήστε ότι δεν υπάρχουν ισοψηφίες.

Μονάδες 8

ΨΗΦΟΣ	1	2	. . .	21	22	← Ψηφίζονται
1	1	1		0	1	
2	1	0		1	1	
Ψηφίζουν → . . .						
21	0	0		1	0	
22	1	0		0	1	

Πρόβλημα 127 / σελίδα 218 / Κώδικας

```
Αλγόριθμος pro_4_127
!Δ1
   Για i από 1 μέχρι 22
      Για j από 1 μέχρι 22
         Αρχή_επανάληψης
            Διάβασε ΨΗΦΟΣ[i, j]
         Μέχρις_ότου ΨΗΦΟΣ[i, j] = 0 ή ΨΗΦΟΣ[i, j] = 1
      Τέλος_επανάληψης
   Τέλος_επανάληψης
!Δ2
   C ← 0
   Για i από 1 μέχρι 22
      S ← 0
      Για j από 1 μέχρι 22
         S ← S + ΨΗΦΟΣ[i, j]
      Τέλος_επανάληψης
      Αν S = 0 τότε
         C ← C + 1
      Τέλος_αν
   Τέλος_επανάληψης
   Εμφάνισε C
!Δ3
   C ← 0
   Για i από 1 μέχρι 22
      Αν ΨΗΦΟΣ[i, i] = 1 τότε
         C ← C + 1
      Τέλος_αν
   Τέλος_επανάληψης
   Εμφάνισε C
```

ΨΗΦΟΣ	1	2	. . .	21	22
1	1	1		0	1
2	1	0		1	1
. . .			...		
21	0	0		1	0
22	1	0		0	1

	1	2	. . .	21	22
SUM	10	12		1	12

	1	2	. . .	21	22
POS	1	2		21	22

```
!Δ4
   Για j από 1 μέχρι 22
      S ← 0
      Για i από 1 μέχρι 22
         S ← S + ΨΗΦΟΣ[i, j]
      Τέλος_επανάληψης
      SUM[j] ← S
      POS[j] ← j
   Τέλος_επανάληψης
   Για i από 2 μέχρι 22
      Για j από 22 μέχρι i με_βήμα -1
         Αν SUM[j - 1] < SUM[j] τότε
            temp ← SUM[j - 1]
            SUM[j - 1] ← SUM[j]
            SUM[j] ← temp
            temp ← POS[j - 1]
            POS[j - 1] ← POS[j]
            POS[j] ← temp
         Τέλος_αν
      Τέλος_επανάληψης
   Τέλος_επανάληψης
   Για i από 1 μέχρι 3
      Εμφάνισε POS[i], SUM[i]
   Τέλος_επανάληψης
Τέλος pro_4_127
```

Αποστάσεις πόλεων / Θέμα με τετραγωνικό, συμμετρικό πίνακα

Το υπουργείο τουρισμού θέλει να ενημερώσει τους τουρίστες σχετικά με τις χιλιομετρικές αποστάσεις μεταξύ 10 πόλεων της Ελλάδας.

Να γραφεί πρόγραμμα το οποίο να:

α. Καταχωρεί σ’ ένα πίνακα ΠΟΛΕΙΣ[10] τα ονόματα 10 πόλεων της Ελλάδας.

β. Καταχωρεί σ’ ένα πίνακα ΑΠΟΣΤΑΣΕΙΣ[10,10] τις αποστάσεις μεταξύ των πόλεων βάζοντας στα στοιχεία της κυρίας διαγωνίου την τιμή 0 (αφού είναι ουσιαστικά η απόσταση μιας πόλης από τον εαυτό της) και σε κάθε στοιχείο την αντίστοιχη χιλιομετρική απόσταση.
Πρέπει να προσέξετε ότι η απόσταση που είναι για παράδειγμα στη θέση ΑΠΟΣΤΑΣΕΙΣ[1,3] είναι η ίδια με την απόσταση στη θέση ΑΠΟΣΤΑΣΕΙΣ[3,1].
Έτσι, κατά την παραπάνω καταχώρηση πρέπει όταν δίνεται η μια απόσταση να καταχωρείται και η αντίστοιχη της.

γ. Διαβάζει τα ονόματα δύο πόλεων και αν και οι δύο υπάρχουν στο πίνακα ΠΟΛΕΙΣ, να εμφανίζει την χιλιομετρική τους απόσταση.
Αν δεν υπάρχει κάποια από τις πόλεις αυτές να τις ζητάει ξανά, μέχρι να βρεθούν και οι δύο.

ΠΟΛΕΙΣ

1	ΑΘΗΝΑ
2	ΒΟΛΟΣ
3	ΛΑΡΙΣΑ
. . .	
10	

ΑΠΟΣΤΑΣΕΙΣ

	1	2	3	. . .	10
1	0	331	355		
2	331	0	60		
3	355	60	0		
. . .					
10					

Ενότητα 4

<u>Ασκήσεις</u>

86 / σελίδα 204

και το θέμα της προηγούμενης σελίδας