

Μάθημα 160

Επανάληψη

B.2

Δομές δεδομένων

Άσκηση 1

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 A [4,5] που θέλουμε να τον διαχειριστούμε ως αραιό αντιπροσωπεύεται από τον μονοδιάστατο B[15].

0	7	0	0	0
1	2	0	0	-3
0	0	4	0	0
0	0	0	0	0

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 M[18] που αναπαριστά 6 μη μηδενικά στοιχεία. Δίνεται ο παρακάτω αλγόριθμος, ο οποίος από τον μονοδιάστατο M[18] δημιουργεί τον αραιό δισδιάστατο Δ[10,20].

Αλγόριθμος αντίστροφος

Δεδομένα // M //

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 ← M[i]

 β ← M[i +**1**.....]

 γ ← M[i +**2**.....]

 Δ[α, β] ← γ

Τέλος_επανάληψης

Αποτελέσματα // Δ //

Τέλος αντίστροφος

Να συμπληρωθούν τα κενά για να λειτουργήσει σωστά ο αλγόριθμος.

Άσκηση 2

Δίνεται το παρακάτω ημιτελές τμήμα αλγορίθμου:

$k \leftarrow 1$

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 4

ΓΙΑ j **ΑΠΟ** 1 **ΜΕΧΡΙ** 5

ΑΝ $\dots \text{ΠΙΝ}[i, j] \neq 0 \dots$ **ΤΟΤΕ**

$A[k] \leftarrow i$

$A[\dots k+1 \dots] \leftarrow \dots j \dots$

$A[\dots k+2 \dots] \leftarrow \dots \text{ΠΙΝ}[i, j] \dots$

$k \leftarrow \dots k+3 \dots$

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Να ξαναγράψετε στο τετράδιό σας το παραπάνω τμήμα αλγορίθμου με τα κενά συμπληρωμένα, έτσι ώστε για τα μη μηδενικά στοιχεία ενός δισδιάστατου πίνακα $\text{ΠΙΝ}[4,5]$ να τοποθετεί σε ένα μονοδιάστατο πίνακα $A[60]$ τις ακόλουθες πληροφορίες: τη γραμμή, τη στήλη, και κατόπιν την τιμή του.

Στοίβα / Μεθοδολογίες 1

Ώθηση

Τμήμα προγράμματος για την εισαγωγή στοιχείου στη στοίβα A[10]:

AN top < 10 **TOTE**

top ← top + 1

ΔΙΑΒΑΣΕ A[top]

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Υπερχείλιση'

ΤΕΛΟΣ_ΑΝ

ή

AN top = 10 **TOTE**

ΓΡΑΨΕ 'Υπερχείλιση'

ΑΛΛΙΩΣ

top ← top + 1

ΔΙΑΒΑΣΕ A[top]

ΤΕΛΟΣ_ΑΝ

Απόθηση

Τμήμα προγράμματος για την εξαγωγή στοιχείου από τη στοίβα A[10]:

AN top > 0 **TOTE**

ΓΡΑΨΕ A[top]

top ← top - 1

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Υποχείλιση'

ΤΕΛΟΣ_ΑΝ

ή

AN top = 0 **TOTE**

ΓΡΑΨΕ 'Υποχείλιση'

ΑΛΛΙΩΣ

ΓΡΑΨΕ A[top]

top ← top - 1

ΤΕΛΟΣ_ΑΝ

Στοίβα / Μεθοδολογίες 2

Γέμισμα στοίβας μέσω επανάληψης

Τμήμα προγράμματος για τη διαδοχική εισαγωγή δέκα τιμών στην στοίβα A[10]:

```
ΟΣΟ top < 10 ΕΠΑΝΑΛΑΒΕ  
  top ← top + 1  
  ΔΙΑΒΑΣΕ A[top]  
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

Άδειασμα στοίβας μέσω επανάληψης

Τμήμα προγράμματος για τη διαδοχική εξαγωγή όλων των τιμών της στοίβας A[10]:

```
ΟΣΟ top > 0 ΕΠΑΝΑΛΑΒΕ  
  ΓΡΑΨΕ A[top]  
  top ← top - 1  
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

Ουρά / Μεθοδολογίες 1

Εισαγωγή

Τμήμα προγράμματος για την εισαγωγή στοιχείου στην ουρά A[10]:

```
AN rear < 10 TOTE
  rear ← rear + 1
  ΔΙΑΒΑΣΕ A[rear]
  AN front = 0 TOTE
    front ← 1
  ΤΕΛΟΣ_ΑΝ
ΑΛΛΙΩΣ
  ΓΡΑΨΕ 'ΓΕΜΑΤΗ ΟΥΡΑ'
ΤΕΛΟΣ_ΑΝ
```

```
AN rear = 10 TOTE
  ΓΡΑΨΕ 'ΓΕΜΑΤΗ ΟΥΡΑ'
ΑΛΛΙΩΣ
  rear ← rear + 1
  ΔΙΑΒΑΣΕ A[rear]
  AN front = 0 TOTE
    front ← 1
  ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΑΝ
```

Εξαγωγή

Τμήμα προγράμματος για την εξαγωγή στοιχείου από την ουρά A[10]:

```
AN front = 0 Η front > rear TOTE
  ΓΡΑΨΕ 'ΑΔΕΙΑ ΟΥΡΑ'
ΑΛΛΙΩΣ
  ΓΡΑΨΕ A[front]
  front ← front + 1
ΤΕΛΟΣ_ΑΝ
```

Αριστερά, εάν μετά από μία εξαγωγή, η ουρά αδειάσει, συνεχίζουμε να εισάγουμε στοιχεία από την τρέχουσα θέση, ενώ δεξιά από την αρχή.

```
AN front = 0 TOTE
  ΓΡΑΨΕ 'ΑΔΕΙΑ ΟΥΡΑ'
ΑΛΛΙΩΣ
  ΓΡΑΨΕ A[front]
  front ← front + 1
  AN front > rear TOTE
    front ← 0
    rear ← 0
  ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΑΝ
```

Ουρά / Μεθοδολογίες 2

Γέμισμα ουράς μέσω επανάληψης

Τμήμα προγράμματος για τη διαδοχική εισαγωγή τιμών στην ουρά A[10], μέχρι να γεμίσει:

```
ΟΣΟ rear < 10 ΕΠΑΝΑΛΑΒΕ
```

```
    rear ← rear + 1
```

```
    ΔΙΑΒΑΣΕ A[rear]
```

```
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

```
ΑΝ front = 0 ΤΟΤΕ
```

```
    front ← 1
```

```
ΤΕΛΟΣ_ΑΝ
```

Άδειαση ουράς μέσω επανάληψης

Τμήμα προγράμματος για τη διαδοχική εξαγωγή όλων των τιμών της ουράς A[10]:

```
ΟΣΟ front > 0 ΚΑΙ front <= rear ΕΠΑΝΑΛΑΒΕ
```

```
    ΓΡΑΨΕ A[front]
```

```
    front ← front + 1
```

```
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

```
front ← 0
```

```
rear ← 0
```

```
ΓΡΑΨΕ 'ΑΔΕΙΑ ΟΥΡΑ'
```

Ουρά / Χρήσιμα

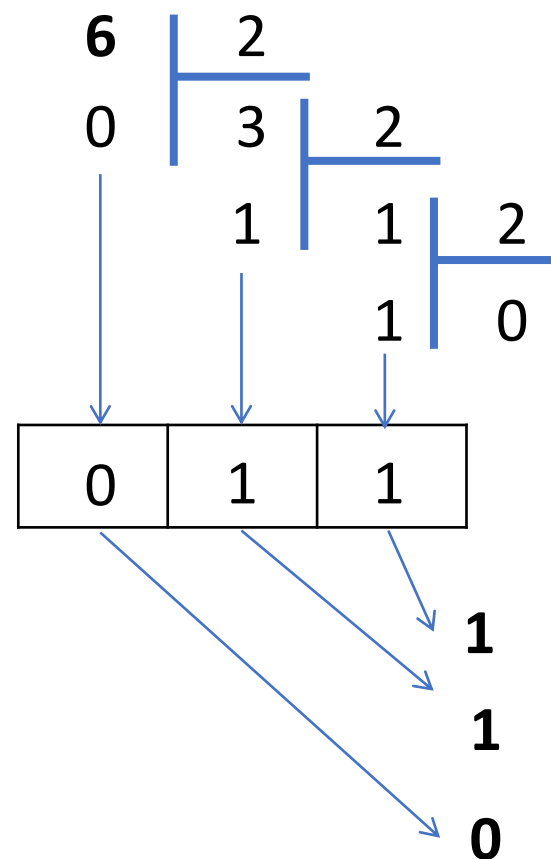
- Η ουρά είναι άδεια όταν οι δείκτες rear και front έχουν τιμή 0, ή όταν ο front, μετά από εξαγωγή, ξεπεράσει τον rear.
- Όταν οι δείκτες rear και front έχουν την ίδια, μη μηδενική τιμή, η ουρά έχει ένα ακριβώς στοιχείο.
- Εάν μετά την εξαγωγή ο front ξεπεράσει τον rear, οπότε και η ουρά αδειάζει, εκχωρούμε και στους δύο την τιμή 0.
- Το πλήθος των στοιχείων μίας όχι άδειας ουράς, είναι $\text{rear} - \text{front} + 1$.

Άσκηση 3

(ΣΑΒΒΑΣ / ΜΕΤΑΤΡΟΠΗ ΔΕΚΑΔΙΚΟΥ ΣΕ ΔΥΑΔΙΚΟ ΜΕ ΣΤΟΙΒΑ)

Να αναπτύξετε πρόγραμμα σε ΓΛΩΣΣΑ το οποίο θα μετατρέπει έναν θετικό ακέραιο στο δυαδικό σύστημα, με τον εξής τρόπο:

- Διαβάζει τον ακέραιο X , ελέγχοντας πως είναι θετικός.
- Διαιρεί τον αριθμό με το 2, ωθεί το υπόλοιπο σε μία στοίβα που υλοποιείται με τον πίνακα $A[10]$ και εκχωρεί το πηλίκο στη μεταβλητή X . Αυτή η διαδικασία γίνεται επαναληπτικά, μέχρι το πηλίκο να γίνει μηδέν. Εάν διαπιστωθεί πως ο πίνακας A δεν επαρκεί για όλα τα δυαδικά ψηφία, η επανάληψη σταματά.
- Απωθεί και εμφανίζει τα στοιχεία της στοίβας A και έτσι προκύπτει ο ζητούμενος δυαδικός, εκτός εάν ο πίνακας A δεν επαρκούσε για τη μετατροπή, οπότε και εμφανίζεται το μήνυμα "ΕΚΤΟΣ ΟΡΙΩΝ".



Άσκηση 3 / κώδικας

ΠΡΟΓΡΑΜΜΑ μάθημα_160_3_ΔΕΚΑΔΙΚΟΣ_ΣΕ_ΔΥΑΔΙΚΟ
ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: X, top, A[10]

ΛΟΓΙΚΕΣ: done

ΑΡΧΗ

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ X

ΜΕΧΡΙΣ_ΟΤΟΥ X > 0

top ← 0

done ← **ΑΛΗΘΗΣ**

ΟΣΟ X > 0 **ΚΑΙ** done = **ΑΛΗΘΗΣ** **ΕΠΑΝΑΛΑΒΕ**

ΑΝ top < 10 **ΤΟΤΕ**

top ← top + 1

A[top] ← X mod 2

X ← X div 2

ΑΛΛΙΩΣ

done ← **ΨΕΥΔΗΣ**

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΝ done = **ΑΛΗΘΗΣ** **ΤΟΤΕ**

ΟΣΟ top > 0 **ΕΠΑΝΑΛΑΒΕ**

ΓΡΑΨΕ A[top]

top ← top - 1

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'ΕΚΤΟΣ ΟΡΙΩΝ'

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Άσκηση 4 / Ουρά και Στοίβα / ΣΑΒΒΑΣ

Να υλοποιηθεί πρόγραμμα που θα διαβάζει έναν αριθμό X . Εάν αυτός είναι ακέραιος, θα τον ωθεί στη στοίβα $S[N]$, διαφορετικά θα τον εισάγει στην ουρά $Q[N]$. Η ουρά και η στοίβα θα υλοποιηθούν με πίνακες, το μέγεθος N των οποίων θα είναι 10 και θα έχει δηλωθεί ως σταθερά. Το πρόγραμμα θα σταματήσει να διαβάζει αριθμούς μόλις μία από τις δύο δομές γεμίσει. Στη συνέχεια θα εμφανίσει πρώτα τα στοιχεία της στοίβας και στη συνέχεια τα στοιχεία της ουράς, καθώς και το πλήθος τους.

ΠΡΟΓΡΑΜΜΑ μάθημα_160_4_ΟΥΡΑ_ΚΑΙ_ΣΤΟΙΒΑ_ΣΑΒΒΑΣ
ΣΤΑΘΕΡΕΣ

$N = 10$

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: top, front, rear, $S[N]$

ΠΡΑΓΜΑΤΙΚΕΣ: X , $Q[N]$

ΑΡΧΗ

top $\leftarrow 0$

front $\leftarrow 0$

rear $\leftarrow 0$

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ X

ΑΝ $X = A_M(X)$ **ΤΟΤΕ**

ΑΝ top $< N$ **ΤΟΤΕ**

top \leftarrow top + 1

$S[top] \leftarrow A_M(X)$

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Η ΣΤΟΙΒΑ ΓΕΜΙΣΕ'

ΤΕΛΟΣ_ΑΝ

ΑΛΛΙΩΣ

ΑΝ rear $< N$ **ΤΟΤΕ**

rear \leftarrow rear + 1

$Q[rear] \leftarrow X$

ΑΝ front = 0 **ΤΟΤΕ**

front $\leftarrow 1$

ΤΕΛΟΣ_ΑΝ

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Η ΟΥΡΑ ΓΕΜΙΣΕ'

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΑΝ

ΜΕΧΡΙΣ_ΟΤΟΥ top = N **Η** rear = N

ΓΡΑΨΕ 'ΠΛΗΘΟΣ ΣΤΟΙΧΕΙΩΝ ΣΤΟΙΒΑΣ: ', top

ΟΣΟ top > 0 **ΕΠΑΝΑΛΑΒΕ**

ΓΡΑΨΕ $S[top]$

top \leftarrow top - 1

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'ΠΛΗΘΟΣ ΣΤΟΙΧΕΙΩΝ ΟΥΡΑΣ: ', rear

ΟΣΟ front \leq rear **ΚΑΙ** front > 0 **ΕΠΑΝΑΛΑΒΕ**

ΓΡΑΨΕ $Q[front]$

front \leftarrow front + 1

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΔΟΥΛΕΙΑ ΓΙΑ ΤΟ ΣΠΙΤΙ

1. Σε μια στοίβα 5 θέσεων έχουν τοποθετηθεί τα στοιχεία: Υ, Λ, Ι, Κ, Ο στην 1η, 2η, 3η, 4η και 5η θέση αντίστοιχα.
 - α. Να προσδιορίσετε την τιμή του δείκτη top της παραπάνω στοίβας και να την σχεδιάσετε.
 - β. Αν εφαρμόσουμε τις λειτουργίες απώθηση, απώθηση, απώθηση, ώθηση Η, ώθηση Σ και απώθηση, ποια είναι η νέα τιμή της top και ποια η τελική μορφή της στοίβας;
2. Σε μια ουρά 10 θέσεων έχουν τοποθετηθεί τα στοιχεία: Π, Α, Λ, Τ, Ο στην 1η, 2η, 3η, 4η και 5η θέση αντίστοιχα.
 - α. Να προσδιορίσετε τις τιμές των δεικτών rear και front της παραπάνω ουράς και να τη σχεδιάσετε.
 - β. Αν εφαρμόσουμε τις ακόλουθες λειτουργίες: Εξαγωγή – Εξαγωγή – Εισαγωγή Μ – Εισαγωγή Η – Εξαγωγή, ποιες είναι τις τιμές των δεικτών rear και front της παραπάνω ουράς και ποια η τελική μορφή της ουράς;
3. Το παρακάτω τμήμα αλγορίθμου υλοποιεί τη λειτουργία της εξαγωγής στοιχείου από ουρά με χρήση μονοδιάστατου πίνακα Α. Συμπληρώστε τα κενά του τμήματος αλγορίθμου, ώστε να επιτελείται η ζητούμενη λειτουργία.

ΑΝ **ΚΑΙ** **ΤΟΤΕ**

ΓΡΑΨΕ 'Άδεια ουρά'

ΑΛΛΙΩΣ_ΑΝ **ΤΟΤΕ**

ΓΡΑΨΕ 'Εξάγεται το στοιχείο:', A[front]

front ← 0

rear ← 0

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Εξάγεται το στοιχείο:', A[front]

.....

ΤΕΛΟΣ_ΑΝ

ΔΟΥΛΕΙΑ ΓΙΑ ΤΟ ΣΠΙΤΙ

4. (ΣΑΒΒΑΣ) Τα smartphones με λειτουργικό Android, διατηρούν μία στοίβα με τις διευθύνσεις των ανοικτών εφαρμογών, δίνοντας τη δυνατότητα, της προς τα πίσω πλοήγησης.
- Ας προσομοιάσουμε κάπως αυτή τη διαδικασία, με ένα πρόγραμμα, το οποίο χρησιμοποιώντας μία στοίβα που υλοποιείται με τον πίνακα $A[100]$:
- α. Θα δίνει τη δυνατότητα τριών επιλογών, οι οποίες θα είναι "R" που θα σημαίνει RUN, "B" που θα σημαίνει BACK και "D" που θα σημαίνει DELETE.
 - β. Κατά την επιλογή του "R" θα ωθείται στη στοίβα το όνομα της εφαρμογής που ανοίγουμε, αφού προηγουμένως ελεγχθεί πως η στοίβα δεν είναι γεμάτη. Εάν είναι γεμάτη, τότε "αρχικοποιείται", τοποθετώντας το όνομα της εφαρμογής στην 1η θέση της στοίβας.
 - γ. Κατά την επιλογή του "B" και εφόσον η στοίβα δεν είναι άδεια θα εμφανίζεται το όνομα της εφαρμογής που κλείνει καθώς και αυτής στην οποία επιστρέφουμε. Εάν η εφαρμογή που κλείνει είναι η τελευταία που έχει απομείνει στη στοίβα, ως όνομα επιστροφής εμφανίζεται κενός χαρακτήρας.
 - δ. Κατά την επιλογή του "D", εμφανίζονται τα ονόματα όλων των ανοικτών εφαρμογών, από την τελευταία που άνοιξε προς την πρώτη και το πρόγραμμα τερματίζεται, αφού πρώτα εμφανιστεί το πλήθος των "αρχικοποιήσεων" που έγιναν στο ερώτημα β.

ΘΕΩΡΙΑ : 25 – 30 / σελίδες 356 – 357