

## 1.2 Ουρά

Κατά την ημέρα αναχώρησης τους από την Αίγινα, οι δύο συμφοιτητές αποφάσισαν σκόπιμα να καθυστερήσουν, προκειμένου να μπουν τελευταίοι (ώστε να βγουν πρώτοι και χωρίς ταλαιπωρία στον Πειραιά) και ταυτόχρονα, να απολαύσουν για περισσότερο χρόνο τις τελευταίες στιγμές των διακοπών τους στο νησί.

Δυστυχώς, όμως, ατύχησαν! Το πλοίο της επιστροφής τους ήταν διαφορετικό. Είχε μία είσοδο στο πίσω μέρος προς επιβίβαση των οχημάτων και μία έξοδο στο μπροστινό μέρος προς αποβίβαση. Έτσι, το αυτοκίνητο που έμπαινε πρώτο, προχωρούσε ευθεία προς την άλλη μεριά του πλοίου και έπαιρνε θέση προς αποβίβαση προς την πόρτα εξόδου. Κάθε αυτοκίνητο που έμπαινε στο πλοίο πάκκαρε υποχρεωτικά πίσω από τα άλλα, που είχαν μπει νωρίτερα, στην ίδια σειρά. Δηλαδή, το αυτοκίνητο που επιβιβαζόταν πρώτο στην Αίγινα θα αποβιβαζόταν πρώτο και στον Πειραιά. Με τον τρόπο αυτό, η επιβίβαση και η αποβίβαση των οχημάτων γινόταν πιο γρήγορα. Η κατασκευή του «γκαράζ» του πλοίου και ο τρόπος στάθμευσης των οχημάτων (τρόπος οργάνωσης των δεδομένων) προσομοιάζε στη δομή της «Ουράς». Δηλαδή, τα δεδομένα που μπαίνουν πρώτα σε μια ουρά είναι αυτά που βγαίνουν και πρώτα.



**Ουρά (Queue)**, ονομάζεται μια δομή δεδομένων το σύνολο των στοιχείων της οποίας είναι διατεταγμένο με τέτοιο τρόπο, ώστε τα στοιχεία που τοποθετήθηκαν πρώτα στην ουρά να λαμβάνονται επίσης πρώτα.

Η παραπάνω μέθοδος ονομάζεται **Πρώτο Μέσα, Πρώτο Έξω ή FIFO** (=First In First Out).

Μπορούμε να φανταστούμε την τοποθέτηση των στοιχείων μιας ουράς σε οριζόντια σειρά. Ο πελάτης μιας τράπεζας που μπαίνει πρώτος σε μια ουρά για να εξυπηρετηθεί, είναι αυτός που εξυπηρετείται και πρώτος, με την έξοδό του από την ουρά αναμονής.

Άλλα παραδείγματα είναι η ουρά στα λεωφορεία ή η ουρά στα ταμεία, όπου ο πρώτος που στέκεται στην ουρά εξυπηρετείται και πρώτος.

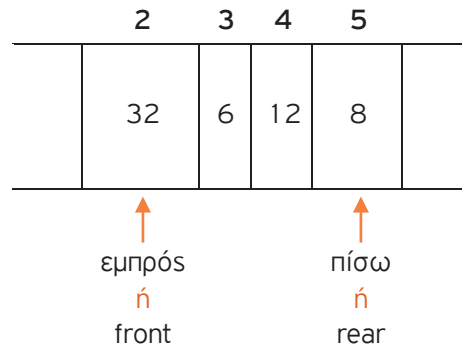


Οι **κύριες λειτουργίες** που εκτελούνται σε μια ουρά είναι δύο:

1. Η **εισαγωγή** (enqueue) στοιχείου στο πίσω άκρο της ουράς.
2. Η **εξαγωγή** (dequeue) στοιχείου από το εμπρός άκρο της ουράς.

## Υλοποίηση ουράς με χρήση μονοδιάστατου πίνακα

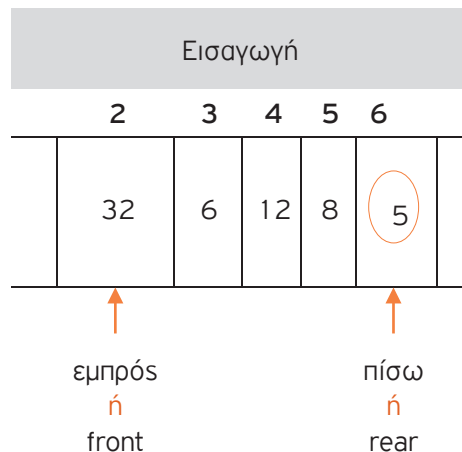
- Χρησιμοποιούμε δύο μεταβλητές, την *front* (ή εμπρός) που δείχνει τη θέση του 1<sup>ου</sup> στοιχείου της ουράς και την *rear* (ή πίσω) που δείχνει τη θέση του τελευταίου στοιχείου. Ως αρχικές τιμές των μεταβλητών *rear* και *front* θεωρούμε το μηδέν.



- Η **εισαγωγή** ενός νέου στοιχείου γίνεται από το πίσω άκρο της ουράς και η τιμή της μεταβλητής *rear* αλλάζει ως εξής:

$$\text{rear} \leftarrow \text{rear} + 1$$

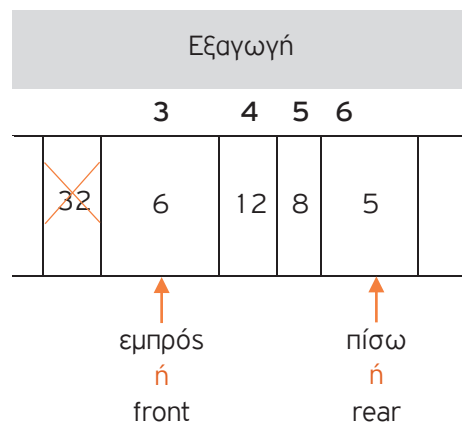
Κατά την εισαγωγή, πρώτα αυξάνουμε τον δείκτη *rear* κατά ένα και μετά εισάγουμε το στοιχείο στον πίνακα.



- Η **εξαγωγή** ενός στοιχείου γίνεται από το εμπρός άκρο της ουράς και η τιμή της μεταβλητής *front* αλλάζει ως εξής:

$$\text{front} \leftarrow \text{front} + 1$$

Κατά την εξαγωγή ενός στοιχείου, αυξάνεται ο δείκτης *front* κατά ένα (δείχνει στην επόμενη θέση του πίνακα) χωρίς στην πραγματικότητα να γίνεται καμία παρέμβαση στα περιεχόμενα του πίνακα (χωρίς να διαγράφεται κάποιο στοιχείο).



### 1.2.1 Παραδείγματα υλοποίησης ουράς με χρήση μονοδιάστατου πίνακα



### Παράδειγμα 1 – Εισαγωγή και Εξαγωγή στοιχείων σε ουρά

1. Σε μια ουρά 10 θέσεων έχουν τοποθετηθεί διαδοχικά τα στοιχεία: Σ, Γ, Μ, Α, Δ στην 1η, 2η, 3η, 4η και 5η θέση αντίστοιχα.
  - i. Να προσδιορίσετε τις τιμές των δεικτών rear και front και να σχεδιάσετε την παραπάνω ουρά.
  - ii. Αν εφαρμόσετε τις παρακάτω λειτουργίες: **Εξαγωγή, Εξαγωγή, Εξαγωγή, Εισαγωγή Χ, Εισαγωγή Δ** και **Εξαγωγή**, ποιες είναι οι τιμές των δεικτών rear και front της ουράς και η τελική μορφή της;
2. Σε μια άδεια ουρά 10 θέσεων εισάγονται τα στοιχεία Ο, Σ, Λ, Τ, Ε. Με ποιον τρόπο πρέπει να εισαχθούν και να εξαχθούν τα στοιχεία, ώστε η έξοδος να εμφανίζει τα στοιχεία Τ, Ε, Λ, Ο, Σ;

### Απάντηση

1)

|              |          |    |    |          |             |    |    |    |     |
|--------------|----------|----|----|----------|-------------|----|----|----|-----|
| 1n           | 2n       | 3n | 4n | 5n       | 6n          | 7n | 8n | 9n | 10n |
| $\Sigma$     | $\Gamma$ | M  | A  | $\Delta$ |             |    |    |    |     |
| <b>front</b> |          |    |    |          | <b>rear</b> |    |    |    |     |

i)

front=1 και rear=5

**ii)** Η τελική μορφή της ουράς είναι:

|    |    |    |    |       |    |      |    |    |     |
|----|----|----|----|-------|----|------|----|----|-----|
| 1n | 2n | 3n | 4n | 5n    | 6n | 7n   | 8n | 9n | 10n |
|    |    |    |    | Δ     | X  | Δ    |    |    |     |
|    |    |    |    | front |    | rear |    |    |     |

και οι τιμές της front και της rear γίνονται: front=5 και rear=7

2)

**i) Η αρχική μορφή της ουράς είναι:**

|              |    |    |    |    |             |    |    |    |     |
|--------------|----|----|----|----|-------------|----|----|----|-----|
| 1n           | 2n | 3n | 4n | 5n | 6n          | 7n | 8n | 9n | 10n |
| O            | Σ  | Λ  | T  | E  |             |    |    |    |     |
| <b>front</b> |    |    |    |    | <b>rear</b> |    |    |    |     |

front=1 and rear=5

ii) Με την εκτέλεση των λειτουργιών: **Εξαγωγή, Εξαγωγή, Εξαγωγή, Εισαγωγή Λ, Εισαγωγή Ο, Εισαγωγή Σ**, η τελική μορφή της ουράς γίνεται:

|              |    |    |    |             |    |    |    |    |     |
|--------------|----|----|----|-------------|----|----|----|----|-----|
| 1n           | 2n | 3n | 4n | 5n          | 6n | 7n | 8n | 9n | 10n |
|              |    |    | T  | E           | Λ  | O  | Σ  |    |     |
| <b>front</b> |    |    |    | <b>rear</b> |    |    |    |    |     |

και οι τιμές της front και της rear γίνονται: front=4 και rear=8



### Παράδειγμα 2 – Εισαγωγή στοιχείου σε ουρά με χρήση μονοδιάστατου πίνακα

Να αναπτύξετε τμήμα προγράμματος σε ΓΛΩΣΣΑ που υλοποιεί την εισαγωγή στοιχείου σε ουρά, με χρήση μονοδιάστατου πίνακα A, 10 θέσεων.

**Απάντηση**



Κώδικας σε ΓΛΩΣΣΑ [1.4]

```

1  ΓΡΑΨΕ 'Δώσε στοιχείο για εισαγωγή στην ουρά A:'
2  ΔΙΑΒΑΣΕ στοιχείο
3  ΑΝ rear = 10 ΤΟΤΕ
4      ΓΡΑΨΕ 'Γεμάτη ουρά'
5  ΑΛΛΙΩΣ_ΑΝ (front = 0 ΚΑΙ rear = 0) ΤΟΤΕ
6      front<- 1
7      rear<-1
8      A[rear]<-στοιχείο
9  ΑΛΛΙΩΣ
10     rear <- rear + 1
11     A[rear]<-στοιχείο
12 ΤΕΛΟΣ_ΑΝ
  
```



### Παράδειγμα 3 – Εξαγωγή στοιχείου από ουρά με χρήση μονοδιάστατου πίνακα

Να αναπτύξετε τμήμα προγράμματος σε ΓΛΩΣΣΑ που υλοποιεί την εξαγωγή στοιχείου από ουρά, με χρήση μονοδιάστατου πίνακα A, 10 θέσεων.

**Απάντηση**



Κώδικας σε ΓΛΩΣΣΑ [1.5]

```

1  ΑΝ (front = 0 ΚΑΙ rear = 0) ΤΟΤΕ
2      ΓΡΑΨΕ 'Άδεια ουρά'
3  ΑΛΛΙΩΣ_ΑΝ (front = rear ) ΤΟΤΕ
4      ΓΡΑΨΕ 'Εξάγεται το στοιχείο:',A[front]
5      front <- 0
6      rear <- 0
7  ΑΛΛΙΩΣ
8      ΓΡΑΨΕ 'Εξάγεται το στοιχείο:',A[front]
9      front <- front + 1
10 ΤΕΛΟΣ_ΑΝ
  
```



#### Παράδειγμα 4 – Επιβίβαση και Αποβίβαση αυτοκινήτων σε πλοίο

Ένα οχηματαγωγό πλοίο με δύο διαφορετικές πόρτες, μία για την είσοδο και μία για την έξοδο των οχημάτων, χωρητικότητας 250 αυτοκινήτων, τα οποία δύνανται να τοποθετηθούν αποκλειστικά σε μία σειρά, εκτελεί το δρομολόγιο ΠΕΙΡΑΙΑΣ – ΑΙΓΙΝΑ. Τα οχήματα που **επιβιβάζονται πρώτα είναι και αυτά που θα αποβιβαστούν πρώτα**. Στο λιμάνι του Πειραιά προσέρχονται τα αυτοκίνητα για αναχώρηση.

Να αναπτύξετε πρόγραμμα σε ΓΛΩΣΣΑ το οποίο:

1. Να υλοποιεί μενού με τις επιλογές:
  1. Επιβίβαση
  2. Αποβίβαση
  3. Έξοδος
2. Στην περίπτωση που επιλεγεί η **Επιβίβαση** το πρόγραμμα θα διαβάζει τον αριθμό κυκλοφορίας καθενός από τα οχήματα που επιβιβάζονται στο πλοίο και θα τον καταχωρίζει στην ουρά ΟΧΗΜΑΤΑ. Κάθε φορά που επιβιβάζεται ένα όχημα να τυπώνεται το ερώτημα «**Υπάρχει όχημα για επιβίβαση; (Ν/Ο)**». Αν ο χρήστης απαντήσει Ν (=ΝΑΙ), τότε να επαναλαμβάνεται η διαδικασία επιβίβασης, ενώ αν απαντήσει Ο (=ΟΧΙ), τότε να σταματά η διαδικασία επιβίβασης και να επιστρέφει το πρόγραμμα στο μενού Επιλογής.
3. Στην περίπτωση που επιλεγεί η **Αποβίβαση** το πρόγραμμα θα εξάγει από την ουρά και θα εμφανίζει όλα τα αυτοκίνητα που αποβιβάστηκαν στην ΑΙΓΙΝΑ.

#### Απάντηση



Παρακάτω υλοποιείται το πρόγραμμα του παραδείγματος επιβίβασης/αποβίβασης σε οχηματαγωγό πλοίο, όπου η «πόρτα» για την είσοδο των οχημάτων στο πλοίο είναι διαφορετική από την «πόρτα» εξόδου των οχημάτων από αυτό.

#### Ανάλυση της λύσης

Σύμφωνα με την εκφώνηση της άσκησης πρέπει:

- Να υλοποιηθεί μια δομή δεδομένων FIFO (ουρά). Η υλοποίηση της ουράς θα πραγματοποιηθεί με μονοδιάστατο πίνακα 250 θέσεων (π[250]) όσο και το μέγιστο πλήθος επιβιβαζόμενων οχημάτων (250).
- Να δηλωθούν δύο βοηθητικές μεταβλητές (δείκτες) που δείχνουν την αρχή («**αρχ**») και το τέλος («**τέλος**») της ουράς. Οι αρχικές τιμές των δεικτών «**αρχ**» και «**τέλος**» θα είναι μηδέν.
- Στην περίπτωση της **επιβίβασης**, πρώτα θα αυξάνεται ο δείκτης «τέλος» κατά 1, για να δείχνει στην επόμενη κενή θέση της ουράς (τέλος<-τέλος+1). Σε αυτή την κενή θέση θα καταχωρίζεται ο αριθμός κυκλοφορίας («**αρ**») του επιβιβαζόμενου οχήματος (π[τέλος]<-αρ).
- Στην περίπτωση της **αποβίβασης**, πρώτα θα αποβιβάζεται το όχημα και μετά ο δείκτης «**αρχ**» θα αυξάνεται κατά 1 (αρχ<-αρχ+1), έτσι ώστε να δείχνει στη θέση που αντιστοιχίζεται στο επόμενο όχημα προς αποβίβαση.
- Να δηλωθούν και οι μεταβλητές: **επ1** (επιλογή μενού), **πλ** (μετρητής αποβιβαζόμενων οχημάτων), **επ2** (μεταβλητή δήλωσης οχήματος για επιβίβαση), **αρ** (αριθμός κυκλοφορίας οχήματος, μέσω του οποίου καταχωρίζεται-δηλώνεται η επιβίβασή του), **π[250]** (πίνακας 250 θέσεων, που φαίνεται το μέγιστο πλήθος των οχημάτων που μπορούν να χωρέσουν στο πλοίο).

- Να εμφανίζεται ένα μενού επιλογών με τις επιλογές: **1.** «Επιβίβαση», **2.** «Αποβίβαση», **3.** «Εξοδος» (δομή επανάληψης «**ΜΕΧΡΙΣ\_ΟΤΟΥ...**», με τη βοήθεια της μεταβλητής «**επ1**» η οποία θα παίρνει τις τιμές 1 ή 2 ή 3).
- Στην περίπτωση της **επιβίβασης** («**ΑΝ επ1 = 1 ΤΟΤΕ**»):
  - ο Αρχικά, γίνεται έλεγχος αν το πλοίο είναι γεμάτο («**ΑΝ τελος=250 ΤΟΤΕ**»). Στην περίπτωση αυτή εμφανίζεται κατάλληλο μήνυμα («**ΓΡΑΨΕ 'Το πλοίο είναι πλήρες και δε χωρά άλλα οχήματα'**»)
  - ο Στη συνέχεια, θα πρέπει να ελεγχθεί αν υπάρχει όχημα για επιβίβαση. Ο έλεγχος πραγματοποιείται μέσω της μεταβλητής «**επ2**» («**ΔΙΑΒΑΣΕ επ2, ΑΝ επ2 <> 'N'...**»).
  - ο Αν υπάρχει όχημα προς επιβίβαση, θα πρέπει να ελέγχεται αν είναι κενό το πλοίο/ουρά. Αν είναι κενό («**ΑΝ (αρχ=0 ΚΑΙ τελος=0) ΤΟΤΕ**»), αυξάνουμε τους δείκτες «**αρχ**» και «**τέλος**» για να δείχνουν στην πρώτη κενή θέση του πλοίου/ουράς («**αρχ<-1, τελος<-1**») και μετά γίνεται καταχώριση του αριθμού κυκλοφορίας του επιβιβαζόμενου οχήματος στη θέση αυτή (πρώτη θέση του πλοίου/ουράς) («**π[τελος]<- αρχ**»).

Στην περίπτωση που το πλοίο/ουρά δεν είναι κενό, αυξάνεται πρώτα η τιμή του πίσω δείκτη της ουράς («**τελος<-τελος+1**»), για να δείχνει σε κενή θέση κι έπειτα καταχωρίζεται σε αυτή ο αριθμός κυκλοφορίας του επιβιβαζόμενου οχήματος. Η διαδικασία αυτή επαναλαμβάνεται όσο υπάρχουν οχήματα προς επιβίβαση και το πλοίο δεν έχει γεμίσει («**ΜΕΧΡΙΣ\_ΟΤΟΥ τελος =250 Η επ2='Ο' Η επ2='ο'**»).

Στην περίπτωση που διακοπεί η επαναληπτική διαδικασία της επιβίβασης λόγω «γεμίσματος» του πλοίου («**ΑΝ τελος=250 ΤΟΤΕ**»), εμφανίζεται κατάλληλο μήνυμα («**ΓΡΑΨΕ 'Το πλοίο είναι πλήρες και δε χωρά άλλα οχήματα'**»).
- Στην περίπτωση της **αποβίβασης** («**ΑΛΛΙΩΣ\_ΑΝ επ1 = 2 ΤΟΤΕ**»):
  - ο Θα πρέπει πρώτα να ελεγχθεί αν το πλοίο/ουρά είναι κενό («**ΑΝ (αρχ=0 ΚΑΙ τελος=0) ΤΟΤΕ**») και στην περίπτωση αυτή να εμφανίζεται κατάλληλο μήνυμα («**ΓΡΑΨΕ 'Το πλοίο είναι άδειο'**»)
  - ο Έπειτα, θα πρέπει να γίνει έλεγχος της περίπτωσης που το πλοίο/ουρά έχει ένα μόνο όχημα («**ΑΛΛΙΩΣ\_ΑΝ αρχ=τελος ΤΟΤΕ**»). Στην περίπτωση αυτή πρώτα αποβιβάζεται το όχημα από το πλοίο («εξάγεται» από την ουρά, «**ΓΡΑΨΕ 'Αποβιβάζεται το μοναδικό όχημα:', π[αρχ]**») και μετά μηδενίζουμε τους δείκτες («**αρχ<-0, 'τέλος<-0**»), αφού το πλοίο/ουρά είναι πλέον κενό.

Σημειώνεται ότι για λόγους παρακολούθησης των τιμών του πίνακα, στη θέση του πίνακα που κατείχε το αποβιβαζόμενο όχημα καταχωρίζεται ο κενός χαρακτήρας («**π[αρχ]<-' '**»).

  - ο Αν υπάρχουν περισσότερα από ένα οχήματα για αποβίβαση, μέσα σε μια επαναληπτική διαδικασία πρώτα εμφανίζεται το όχημα που αποβιβάζεται («**ΓΡΑΨΕ 'Αποβιβάζεται το όχημα:', π[αρχ]**») κι έπειτα ο μπροστινός δείκτης «**αρχ**» αυξάνεται κατά 1 («**αρχ<-αρχ+1**») και δείχνει στη θέση του επόμενου οχήματος προς αποβίβαση. Η επαναληπτική διαδικασία αποβίβασης ολοκληρώνεται, όταν πλέον δεν υπάρχει κανένα όχημα για αποβίβαση, δηλαδή όταν ο δείκτης «**αρχ**» γίνει μεγαλύτερος από τον πίσω δείκτη «**τέλος**» της ουράς («**ΜΕΧΡΙΣ\_ΟΤΟΥ (αρχ>τελος)**»).
  - ο Τέλος, με την ολοκλήρωση της αποβίβασης όλων των οχημάτων, εκτυπώνεται με κατάλληλο μήνυμα το πλήθος αποβιρασθέντων οχημάτων στην ΑΙΓΙΝΑ («**ΓΡΑΨΕ 'Οχήματα που αποβιβάστηκαν στην ΑΙΓΙΝΑ: ', πλ**»).

- Ο Σε κάθε νέα εκκίνηση αποβίβασης οχημάτων, ο μετρητής «πλ» πρέπει να μηδενίζεται, για να είναι σωστό το σύνολο των αποβιbasθέντων οχημάτων της τρέχουσας αποβίbasης. Π.χ. εκτελείται το πρόγραμμα και επιβιβάζονται 15 οχήματα, τα οποία στη συνέχεια αποβιβάζονται. Στη συνέχεια, επιλέγεται, από το μενού, νέα επιβίbasη 30 οχημάτων, τα οποία στη συνέχεια αποβιβάζονται. Αν ο μετρητής «πλ» δεν μηδενίζεται σε κάθε νέα εκκίνηση αποβίbasης, στην πρώτη αποβίbasη θα είχε την τιμή «15» και στη δεύτερη την τιμή «45» και όχι την τιμή «30» που είναι η σωστή. Δηλαδή, στη δεύτερη αποβίbasη οχημάτων το πρόγραμμα θα τύπωνε λανθασμένα «Οχήματα που αποβιbasήθηκαν στην ΑΙΓΙΝΑ: 45» και όχι το σωστό «Οχήματα που αποβιbasήθηκαν στην ΑΙΓΙΝΑ: 30» («**ΓΡΑΨΕ** 'Οχήματα που αποβιbasήθηκαν στην ΑΙΓΙΝΑ: ', πλ»).
- Επίσης, αν επιλεχθεί νέα επιβίbasη μετά την ολοκλήρωση κάθε αποβίbasης, θα πρέπει η ουρά να γεμίζει από την αρχή. Κάτι τέτοιο επιτυγχάνεται με τον μηδενισμό των τιμών των δύο δεικτών, καθώς παίρνουν την ίδια τιμή («αρχ <- 0», «τελος <- 0»).

Ακολουθεί η υλοποίηση του προγράμματος σε ΓΛΩΣΣΑ.



#### Κώδικας σε ΓΛΩΣΣΑ [1.6]

**ΠΡΟΓΡΑΜΜΑ** πλοίο2

**ΜΕΤΑΒΛΗΤΕΣ**

**ΑΚΕΡΑΙΕΣ:** αρχ, τελος, επ1, πλ

**ΧΑΡΑΚΤΗΡΕΣ:** επ2, αρ, π[250]

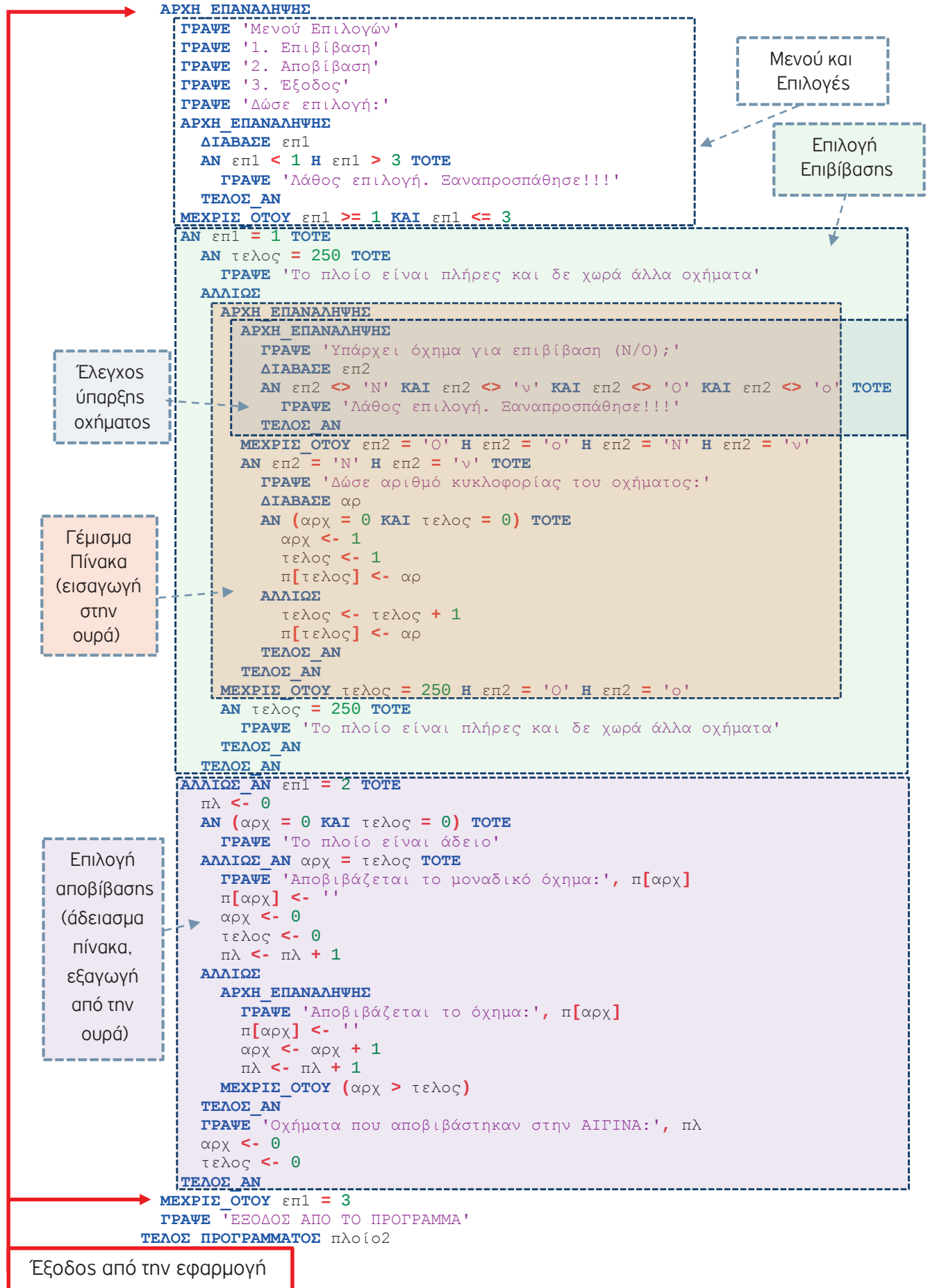
**ΑΡΧΗ**

αρχ <- 0

τελος <- 0

Δήλωση Μεταβλητών

Αρχικοποίηση  
Μεταβλητών



```

1  ΠΡΟΓΡΑΜΜΑ πλοίο2
2  ΜΕΤΑΒΛΗΤΕΣ
3      ΑΚΕΡΑΙΕΣ: αρχ, τελος, επ1, πλ
4      ΧΑΡΑΚΤΗΡΕΣ: επ2, αρ, π[250]
5  ΑΡΧΗ
6      αρχ <- 0
7      τελος <- 0
8      ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
9          ΓΡΑΨΕ 'Μενού Επιλογών'
10         ΓΡΑΨΕ '1. Επιβίβαση'
11         ΓΡΑΨΕ '2. Αποβίβαση'
12         ΓΡΑΨΕ '3. Έξοδος'
13         ΓΡΑΨΕ 'Δώσε επιλογή:'
14         ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
15             ΔΙΑΒΑΣΕ επ1
16             ΑΝ επ1 < 1 Η επ1 > 3 ΤΟΤΕ
17                 ΓΡΑΨΕ 'Λάθος επιλογή. Ξαναπροσπάθησε!!!'
18             ΤΕΛΟΣ_ΑΝ
19             ΜΕΧΡΙΣ_ΟΤΟΥ επ1 >= 1 ΚΑΙ επ1 <= 3
20             ΑΝ επ1 = 1 ΤΟΤΕ
21                 ΑΝ τελος = 250 ΤΟΤΕ
22                     ΓΡΑΨΕ 'Το πλοίο είναι πλήρες και δε χωρά άλλα οχήματα'
23                 ΑΛΛΙΩΣ
24                     ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
25                         ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
26                             ΓΡΑΨΕ 'Υπάρχει όχημα για επιβίβαση (N/O);'
27                             ΔΙΑΒΑΣΕ επ2
28                             ΑΝ επ2<>'N' ΚΑΙ επ2<>'ν' ΚΑΙ επ2<>'Ο' ΚΑΙ επ2<>'ο' ΤΟΤΕ
29                                 ΓΡΑΨΕ 'Λάθος επιλογή. Ξαναπροσπάθησε!!!'
30                             ΤΕΛΟΣ_ΑΝ
31                             ΜΕΧΡΙΣ_ΟΤΟΥ επ2='Ο' Η επ2='ο' Η επ2='N' Η επ2='ν'
32                             ΑΝ επ2 = 'N' Η επ2 = 'ν' ΤΟΤΕ
33                                 ΓΡΑΨΕ 'Δώσε αριθμό κυκλοφορίας του οχήματος:'
34                                 ΔΙΑΒΑΣΕ αρ
35                                 ΑΝ (αρχ = 0 ΚΑΙ τελος = 0) ΤΟΤΕ
36                                     αρχ <- 1
37                                     τελος <- 1
38                                     π[τελος] <- αρ
39                                 ΑΛΛΙΩΣ
40                                     τελος <- τελος + 1

```

```

41         π[τελος] <- αρ
42         ΤΕΛΟΣ_ΑΝ
43         ΤΕΛΟΣ_ΑΝ
44         ΜΕΧΡΙΣ_ΟΤΟΥ τελος = 250 Η επ2 = 'Ο' Η επ2 = 'ο'
45         ΑΝ τελος = 250 ΤΟΤΕ
46             ΓΡΑΨΕ 'Το πλοίο είναι πλήρες και δε χωρά άλλα οχήματα'
47         ΤΕΛΟΣ_ΑΝ
48         ΤΕΛΟΣ_ΑΝ
49     ΑΛΛΙΩΣ_ΑΝ επ1 = 2 ΤΟΤΕ
50         πλ <- 0
51         ΑΝ (αρχ = 0 ΚΑΙ τελος = 0) ΤΟΤΕ
52             ΓΡΑΨΕ 'Το πλοίο είναι άδειο'
53         ΑΛΛΙΩΣ_ΑΝ αρχ = τελος ΤΟΤΕ
54             ΓΡΑΨΕ 'Αποβιβάζεται το μοναδικό όχημα:', π[αρχ]
55             π[αρχ] <- ''
56             αρχ <- 0
57             τελος <- 0
58             πλ <- πλ + 1
59         ΑΛΛΙΩΣ
60             ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
61                 ΓΡΑΨΕ 'Αποβιβάζεται το όχημα:', π[αρχ]
62                 π[αρχ] <- ''
63                 αρχ <- αρχ + 1
64                 πλ <- πλ + 1
65             ΜΕΧΡΙΣ_ΟΤΟΥ (αρχ > τελος)
66             ΤΕΛΟΣ_ΑΝ
67             ΓΡΑΨΕ 'Οχήματα που αποβιβάστηκαν στην ΑΙΓΙΝΑ:', πλ
68             αρχ <- 0
69             τελος <- 0
70             ΤΕΛΟΣ_ΑΝ
71         ΜΕΧΡΙΣ_ΟΤΟΥ επ1 = 3
72         ΓΡΑΨΕ 'ΕΞΟΔΟΣ ΑΠΟ ΤΟ ΠΡΟΓΡΑΜΜΑ'
73     ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ πλοίο2

```

## 1.2.2 Ερωτήσεις - Ασκήσεις

**Ε. 1:** Να δώσετε παραδείγματα ουράς από την καθημερινή ζωή.

**Ε. 2:** Δίνεται η επόμενη ακολουθία αριθμών : 4, 8, 2, 5, 9, 13.

1. Ποια λειτουργία θα χρησιμοποιήσετε για την τοποθέτηση των αριθμών σε ουρά;
2. Να σχεδιάσετε την ουρά έπειτα από την τοποθέτηση των αριθμών.
3. Ποια λειτουργία θα χρησιμοποιήσετε για την εξαγωγή των αριθμών από την ουρά;
4. Πόσες φορές θα πρέπει να εκτελεστεί η προηγούμενη λειτουργία στην ουρά για να εξαχθεί ο αριθμός 5;

**Ε. 3:**

1. Σε μια ουρά 10 θέσεων έχουν τοποθετηθεί διαδοχικά τα στοιχεία: X, A, B, A, P στην 1η, 2η, 3η, 4η και 5η θέση αντίστοιχα.
  - i. Να προσδιορίσετε τις τιμές των δεικτών rear και front της παραπάνω ουράς και να τη σχεδιάσετε.
  - ii. Αν εφαρμόσουμε τις ακόλουθες λειτουργίες: **Εξαγωγή, Εξαγωγή, Εξαγωγή, Εισαγωγή X, Εισαγωγή Δ** και **Εξαγωγή** ποιες είναι τις τιμές των δεικτών rear και front της παραπάνω ουράς και ποια η τελική μορφή της ουράς;
2. Σε μια κενή ουρά 10 θέσεων εισάγουμε τα στοιχεία K, Φ, Ι, Α,Ρ. Με ποιον τρόπο πρέπει να «εισαχθούν» και να «εξαχθούν» τα στοιχεία, ώστε να έχουμε ως έξοδο τα δεδομένα A, P, X, H.

**Ε. 4:** Σε μια τράπεζα χρησιμοποιείται αυτόματο ηλεκτρονικό μηχανήμα που το χειρίζονται οι πελάτες, οι ταμίες και ο διευθυντής της τράπεζας. Κάθε ένας από τους χειριστές του μηχανήματος έχει δικαιώματα χρήσης συγκεκριμένων πλήκτρων του πληκτρολογίου.

Ο πελάτης το «Π», οι ταμίες το «1» ή το «2» ή το «3» ή το «4» αναλόγως της θέσης του ταμείου που εργάζονται και ο διευθυντής το πλήκτρο «Δ». Κατά την είσοδό του, ο κάθε πελάτης πατάει το πλήκτρο «Π» και εκτυπώνεται ένα χαρτί, στο οποίο αναγράφεται το νούμερο που έχει στην ουρά από την αρχή της ημέρας.

Η τράπεζα έχει 4 ταμεία, όπου όταν ο ταμίας εξυπηρετεί έναν πελάτη, πατάει το νούμερο του ταμείου του, «1» ή «2» ή «3» ή «4».

Ο διευθυντής της τράπεζας, πατώντας το κουμπί «Δ», σταματά τη διαδικασία εξυπηρέτησης των πελατών και μπορεί να δει το σύνολο των πελατών που έχουν ήδη εξυπηρετηθεί από το κάθε ταμείο. Ο μέγιστος αριθμός πελατών που μπορεί να εξυπηρετήσει η τράπεζα είναι 1.000 πελάτες.

Λαμβάνοντας υπόψη ότι ο πρώτος πελάτης εξυπηρετείται πρώτος και ο τελευταίος εξυπηρετείται τελευταίος, να αναπτύξετε πρόγραμμα σε ΓΛΩΣΣΑ, όπου:

1. Να υπάρχει μενού επιλογής:  
**Π. Πελάτης      Τ. Ταμίας      Δ. Διευθυντής**
  2. Στην περίπτωση που επιλεγθεί από το μενού το **Π. Πελάτης**, το πρόγραμμα εκτυπώνει το νούμερο που έχει στην ουρά (από την αρχή της ημέρας).
  3. Στην περίπτωση που επιλεγθεί από το μενού το **Τ. Ταμίας**, ο/η αρμόδιος/-α υπάλληλος επιλέγει το νούμερο του ταμείου που του/της αντιστοιχεί: «**1**» ή «**2**» ή «**3**» ή «**4**» και ο πελάτης διαγράφεται από την ουρά.
  4. Στην περίπτωση που επιλεγθεί από το μενού το **Δ. Διευθυντής**, σταματά η διαδικασία εξυπηρέτησης και το πρόγραμμα τυπώνει το νούμερο του ταμείου που εξυπηρέτησε τους περισσότερους πελάτες.
- Στο πρόγραμμα να γίνεται έλεγχος των δεδομένων εισόδου.

**Ε. 5:** Μια αεροπορική εταιρεία εκτελεί το δρομολόγιο Αθήνα – Θεσσαλονίκη κατά την περίοδο του Σεπτεμβρίου. Λόγω της Δ.Ε.Θ. υπάρχει αυξημένη ζήτηση και η εταιρεία διατηρεί λίστα αναμονής για τους επιβάτες που δεν πρόλαβαν να κλείσουν εισιτήριο, ώστε αν προκύψει κάποια ακύρωση, να ενημερώσει τον πρώτο στη σειρά πελάτη που εισήχθη στη λίστα αναμονής προκειμένου να κλείσει εισιτήριο. Η λίστα αναμονής δεν μπορεί να περιλαμβάνει περισσότερα από 10 ονόματα.

Να αναπτύξετε πρόγραμμα σε ΓΛΩΣΣΑ το οποίο:

1. Να υπάρχει μενού επιλογής: **1. ΕΓΓΡΑΦΗ 2. ΑΚΥΡΩΣΗ 3. ΤΕΛΟΣ**.
2. Αν ο χρήστης επιλέξει την τιμή «**1.ΕΓΓΡΑΦΗ**», τότε θα ζητείται το όνομα του χρήστη και θα καταχωρίζεται στη λίστα αναμονής, εφόσον η λίστα αναμονής δεν έχει γεμίσει. Διαφορετικά, θα εμφανίζεται το μήνυμα: «Η λίστα αναμονής είναι πλήρης».
3. Αν ο χρήστης επιλέξει την τιμή «**2.ΑΚΥΡΩΣΗ**», τότε κάποιος από τους επιβάτες της πτήσης έχει ακυρώσει την κράτησή του, συνεπώς, το πρόγραμμα θα πρέπει να εμφανίσει το όνομα του ατόμου που είναι το πρώτο διαθέσιμο στη λίστα αναμονής. Αν δεν υπάρχουν άτομα στη λίστα αναμονής, εμφανίζεται το μήνυμα «Η λίστα αναμονής είναι άδεια».
4. Η παραπάνω διαδικασία επαναλαμβάνεται μέχρι ο χρήστης να επιλέξει την τιμή «**3.ΤΕΛΟΣ**». Το πρόγραμμα εμφανίζει το πλήθος των ατόμων που κατάφεραν να κάνουν κράτηση μέσα από την λίστα αναμονής, καθώς και το μέγιστο πλήθος των ατόμων που περίμεναν στην ουρά αναμονής.

Στο πρόγραμμα να γίνεται έλεγχος εγκυρότητας των τιμών που πληκτρολογούνται.

**Ε. 6:** Σε ένα ταχυδρομικό κατάστημα, οι πελάτες εξυπηρετούνται με βάση τη σειρά άφιξής τους σε αυτό. Το ταχυδρομικό κατάστημα έχει ένα ταμείο και ο μέσος χρόνος εξυπηρέτησης κάθε πελάτη είναι 3 λεπτά. Η ουρά αναμονής στο κατάστημα δεν μπορεί να ξεπερνά τα 30 άτομα.

Να αναπτύξετε πρόγραμμα σε ΓΛΩΣΣΑ το οποίο:

1. Να δέχεται σαν είσοδο από τον χρήστη μία εκ των δύο τιμών εισαγωγής: «**1.ΕΙΣΑΓΩΓΗ**» ή «**2.ΕΠΟΜΕΝΟΣ**» (με έλεγχο εγκυρότητας).

2. Αν δοθεί η τιμή «**1.ΕΙΣΑΓΩΓΗ**», τότε το πρόγραμμα να διαβάζει το ονοματεπώνυμο του πελάτη και αμέσως μετά να εμφανίζει το πλήθος των ατόμων που περιμένουν πριν από αυτόν, εκτός αν η ουρά αναμονής είναι γεμάτη, οπότε να εμφανίζει το μήνυμα «Το κατάστημα γέμισε. Παρακαλούμε ελάτε άλλη φορά».
3. Αν δοθεί η τιμή «**2.ΕΠΟΜΕΝΟΣ**», τότε το πρόγραμμα να εμφανίζει το ονοματεπώνυμο του πελάτη προς εξυπηρέτηση.
4. Η παραπάνω διαδικασία να επαναλαμβάνεται μέχρι να εξυπηρετηθούν όλοι οι πελάτες.
5. Στο τέλος το πρόγραμμα να εμφανίζει το πλήθος των ατόμων που εξυπηρετήθηκαν, καθώς και τον μέσο χρόνο αναμονής των πελατών.

**Ε. 7:** Ένας εκτυπωτής χρησιμοποιεί μια ουρά εκτύπωσης για να τοποθετεί σε αυτήν τα αρχεία που έχουν σταλεί προς εκτύπωση με τη σειρά που αυτά στάλθηκαν. Κάθε φορά εκτυπώνει το αρχείο που βρίσκεται στην αρχή της ουράς εκτύπωσης, το οποίο και εξάγει. Λόγω της περιορισμένης μνήμης του εκτυπωτή, θεωρούμε ότι στην ουρά μπορούν να εισαχθούν το πολύ 15 αρχεία.

Να αναπτύξετε πρόγραμμα σε ΓΛΩΣΣΑ το οποίο:

1. Να διαβάζει επαναληπτικά, με έλεγχο εγκυρότητας, το γράμμα “N” που καθορίζει την έλευση νέου αρχείου ή το γράμμα “E” που δηλώνει την προσπάθεια εκτύπωσης ενός αρχείου.
2. Κατά την έλευση ενός αρχείου, διαβάζει το όνομά του και εξετάζει αν υπάρχει ο διαθέσιμος χώρος στην ουρά και το αρχείο καταχωρίζεται σε αυτήν με τη διαδικασία της εισαγωγής. Στην περίπτωση που δεν υπάρχει χώρος, εμφανίζεται το μήνυμα «Η ουρά γέμισε. Δε μπορεί να εκτυπωθεί το αρχείο».
3. Όταν ο χρήστης δώσει το γράμμα “E”, εξετάζει αν υπάρχουν αρχεία προς εκτύπωση και στην περίπτωση αυτή εξάγεται το κατάλληλο αρχείο εμφανίζοντας τη λέξη «Εκτύπωση» ακολουθούμενη από το όνομα του αρχείου που τυπώνεται.
4. Η επαναληπτική διαδικασία ολοκληρώνεται, όταν εκτυπωθούν όλα τα αρχεία που έχουν τοποθετηθεί στην ουρά.
5. Μετά το τέλος της διαδικασίας, το πρόγραμμα εμφανίζει τον συνολικό αριθμό των αρχείων που εκτυπώθηκαν.

**Ε. 8:** Χαρακτηρίστε τις παρακάτω προτάσεις ως Σωστές ή Λάθος. Στην περίπτωση που πιστεύετε ότι είναι λανθασμένες δικαιολογήστε την επιλογή σας και σκεφτείτε ποια θα μπορούσε να είναι η αντίστοιχη σωστή πρόταση.

| α/α | Προτάσεις                                                                                                 | Σ                        | Λ                        |
|-----|-----------------------------------------------------------------------------------------------------------|--------------------------|--------------------------|
| 1   | Για την υλοποίηση της ουράς μπορεί να χρησιμοποιηθεί πίνακας.                                             | <input type="checkbox"/> | <input type="checkbox"/> |
| 2   | Κατά την εισαγωγή ενός στοιχείου σε ουρά, αυτό τοποθετείται στο μπροστινό άκρο της.                       | <input type="checkbox"/> | <input type="checkbox"/> |
| 3   | Σε μια ουρά κάθε στοιχείο της εξάγεται από το μπροστινό άκρο της.                                         | <input type="checkbox"/> | <input type="checkbox"/> |
| 4   | Η απώθηση είναι μια από τις λειτουργίες της ουράς.                                                        | <input type="checkbox"/> | <input type="checkbox"/> |
| 5   | Η εισαγωγή και η εξαγωγή είναι οι δύο βασικές λειτουργίες της ουράς.                                      | <input type="checkbox"/> | <input type="checkbox"/> |
| 6   | Στην ουρά το στοιχείο που μπαίνει πρώτο βγαίνει και πρώτο.                                                | <input type="checkbox"/> | <input type="checkbox"/> |
| 7   | Η υλοποίηση της ουράς χρησιμοποιεί μία μεταβλητή-δείκτη για την εκτέλεση των δύο βασικών λειτουργιών της. | <input type="checkbox"/> | <input type="checkbox"/> |
| 8   | Η λειτουργία της εξαγωγής μπορεί να εκτελεστεί σε μια γεμάτη ουρά.                                        | <input type="checkbox"/> | <input type="checkbox"/> |