

## 8η Σειρά Ασκήσεων

Οι παρακάτω ασκήσεις προορίζονται για εξάσκηση στη δυναμική διαχείριση μνήμης και τις δυναμικές δομές δεδομένων (π.χ. συνδεδεμένες λίστες, δέντρα). Το αυτόματο σύστημα υποβολής δεν ελέγχει κατά πόσο πραγματικά τις έχετε υλοποιήσει χρησιμοποιώντας δυναμικές δομές δεδομένων. Η υποβολή μπορεί να γίνεται από οποιονδήποτε υπολογιστή, και εκτός εργαστηρίου.

### Άσκηση 18.

Εστω ο **αφηρημένος τύπος δεδομένων** `queue` για την παράσταση μιας ουράς ακεραίων αριθμών, ο οποίος υποστηρίζει μόνο τις ακόλουθες πράξεις:

```
queue empty    (void);           /* επιστρέφει μία κενή ουρά */
int  isEmpty (queue q);         /* ελέγχει αν η ουρά είναι κενή */
void enqueue (queue *q, int x); /* εισάγει ένα στοιχείο στην ουρά */
int  dequeue (queue *q);         /* αφαιρεί ένα στοιχείο από μια μη κενή ουρά */
int  peek    (queue q);         /* επιστρέφει το πρώτο στοιχείο μιας μη κενής ουράς,
                                   χωρίς να το αφαιρέσει */
```

Υλοποιήστε αυτόν τον αφηρημένο τύπο δεδομένων χρησιμοποιώντας συνδεδεμένη λίστα.

Στη συνέχεια, χρησιμοποιήστε τον για να γράψετε ένα πρόγραμμα που να διαβάζει μία ακολουθία μη μηδενικών ακεραίων αριθμών (οσοδήποτε πολλών) και να ελέγχει αν αυτή:

- 1) έχει τόσους θετικούς αριθμούς όσους και αρνητικούς, και (συγχρόνως)
- 2) οι αρνητικοί αριθμοί εμφανίζονται με την ίδια σειρά κατ' απόλυτο τιμή που εμφανίζονται και οι θετικοί αριθμοί.

Αν αυτό συμβαίνει, το πρόγραμμα πρέπει να εκτυπώνει το μήνυμα “yes”, διαφορετικά να εκτυπώνει το μήνυμα “no”.

Παραδείγματα εισόδου:

```
3 -3 -4 -6 4 -7 6 7 | 3 -3 4 -6 -4 -7 6 7
```

Παραδείγματα εξόδου:

```
yes | no
```

► Μπορεί να υποβληθεί στο αυτόματο σύστημα υποβολής και ελέγχου