

# ΠΡΟΧΩΡΗΜΕΝΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ Ι :

## ΕΙΣΑΓΩΓΗ ΣΤΗ C++

### Θέματα Εξετάσεων Ιανουαρίου 2009

1. *Πρώτος αριθμός* λέγεται κάθε θετικός ακέραιος που διαιρείται ακριβώς μόνο από το 1 και τον εαυτό του. *Αριθμός Mersenne* λέγεται ο ακέραιος που μπορεί να γραφτεί στη μορφή  $2^n - 1$ , για κάποιο  $n$ . Να βρείτε και να τυπώσετε στο αρχείο "mersenne.dat" όλους τους ακεραίους αριθμούς μέχρι το 50000 που είναι ταυτόχρονα πρώτοι και Mersenne. Η εκτύπωση κάθε αριθμού θα γίνεται σε ξεχωριστή γραμμή. 25/100
2. Να υλοποιήσετε σε συνάρτηση τον αλγόριθμο της *δυναδικής αναζήτησης* ενός στοιχείου σε έναν *ήδη ταξινομημένο* πίνακα με αριθμούς οποιουδήποτε τύπου. Στον αλγόριθμο αυτό γίνεται αρχικά η επιλογή του μεσαίου στοιχείου (ή του πλησιέστερου στη μέση) και η σύγκρισή του με το ζητούμενο. Το αποτέλεσμα της σύγκρισης καθορίζει σε ποιο "μισό" τμήμα του πίνακα μπορεί να βρίσκεται το επιθυμητό στοιχείο. Επαναλαμβάνουμε τη διαδικασία για το συγκεκριμένο τμήμα (δηλ. συγκρίνουμε το μεσαίο του νέου τμήματος με το ζητούμενο) έτσι ώστε διαδοχικά να περιορίζουμε το διάστημα που είναι πιθανό να υπάρχει το στοιχείο που αναζητούμε. Η διαδικασία σταματά όταν το διάστημα αυτό συρρικνωθεί τόσο ώστε να γνωρίζουμε με βεβαιότητα την ύπαρξη ή μη του ζητούμενου.  
Η συνάρτηση που θα γράψετε θα δέχεται ως ορίσματα τον πίνακα, το πλήθος των στοιχείων του και την ζητούμενη τιμή και θα επιστρέφει (με όποιον τρόπο επιλέξετε) (α) το αν βρέθηκε το στοιχείο και (β) σε ποια *θέση* στον πίνακα υπάρχει (αν υπάρχει). 35/100
3. Υλοποιήστε με αναδρομική συνάρτηση τον αλγόριθμο ταξινόμησης quick-sort. Η ρουτίνα θα δέχεται ως όρισμα ένα μονοδιάστατο vector της STL, με στοιχεία οποιουδήποτε τύπου, το οποίο θα ταξινομεί. 40/100  
Η απλή εκδοχή του αλγορίθμου είναι η ακόλουθη:
  - (α) Επιλέγουμε ένα οποιοδήποτε στοιχείο της αρχικής συλλογής (π.χ. το μεσαίο).
  - (β) Δημιουργούμε ένα νέο, *κατάλληλο* container της STL, με μοναδικό στοιχείο ένα αντίγραφο του επιλεγμένου στο προηγούμενο βήμα.
  - (γ) Αντιγράφουμε (με οποιαδήποτε σειρά) σε διαδοχικές θέσεις *πριν* το μοναδικό στοιχείο στο νέο container τα στοιχεία του αρχικού vector που είναι *μικρότερα* από το επιλεγμένο στοιχείο.
  - (δ) Αντιγράφουμε (με οποιαδήποτε σειρά) σε διαδοχικές θέσεις *μετά* το αρχικό στοιχείο στο νέο container τα στοιχεία του αρχικού vector που είναι *μεγαλύτερα ή ίσα* από το επιλεγμένο στοιχείο.
  - (ε) Αντιγράφουμε τα στοιχεία του νέου container στον παλιό (σβήνοντας τα δικά του).

(γ) Επαναλαμβάνουμε τη διαδικασία για τους δύο υπο-vectors εκατέρωθεν του επιλεγμένου στοιχείου, *αν έχουν, βέβαια, διάσταση μεγαλύτερη από 1* (αλλιώς δεν χρειάζονται ταξινόμηση).

Χρησιμοποιήστε τη ρουτίνα σας για να ταξινομήσετε τα στοιχεία  $(-1.2, 9.1, 8.2, -6.4, 3.8, 0.34, 5.67, 8.99, -9.01, 0.001, 4.8, -3.8, 5.67)$ .