ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ 2020-2021 1η ΕΡΓΑΣΙΑ

ΚΟΤΡΩΤΣΟΣ ΣΩΤΗΡΙΟΣ 3180085

**ΑΝΑΦΟΡΑ ΠΑΡΑΔΟΣΗΣ:**

a.

Για την υλοποιήση του πρώτου ερωτήματος χρησιμοποιούμε τη κλάση Node με ορίσματα data , next , prev , όπου κρατάμε τα δεδομένα και δείχνουμε στον προηγούμενο και τον επόμενο κόμβο.

Αρχικά ορίσαμε δύο μεταβλητές για τα άκρα ( head και tail) της λίστας καθώς και ένα μετρητή που αυξάνεται η μειώνεται ανάλογα με τη μέθοδο που πραγματοποιείται (π.χ. αυξάνεται με την addFirst() ενώ μειώνεται με την removeLast()).

Χρησιμοποιούμε τις μέθοδους:

- isEmpty() για να ελέγχουμε αν υπάρχουν στοιχεία στη λίστα. Οι μέθοδοι εισαγωγής και εξαγωγής πραγματοποιούνται σε χρόνο O(1) καθώς δεν υπάρχει κάποια μορφή σάρωσης των στοιχείων της λίστας με χρήση επαναληπτικών μεθόδων , αλλά επιλέγεται ποιο στοιχείο θα εισαχθεί/διαγραφεί χρησιμοποιώντας τα δύο άκρα της λίστας τα οποία είναι ήδη ορισμένα.
- addFirst() και addLast() που προσθέτουν τον κόμβο που δέχονται ως όρισμα στην αρχή ή στο τέλος αντίστοιχα.
- removeFirst() και removeLast() που διαγράφουν τον κόμβο που βρίσκεται στην αρχή ή στο τέλος αντίστοιχα και επιστρέφουν τα δεδομένα του.
- getFirst() και getLast() επιστρέφουν τα δεδομένα της αρχής και τους τέλους της λίστας.
- printQueue εκτυπώνει τα δεδομένα της λίστας μας
- size() επιστρέφει ακέραια μεταβλητή και συγκεκριμένα τον μετρητή count που είναι η ακέραια μεταβλητή που εκφράζει το ακριβές μέγεθος της λίστας. Οπότε πάλι αποφεύγεται η σάρωση της τελικής λίστας και έτσι ο χρόνος της είναι επίσης O(1).

b.

Ο κώδικας δέχεται σαν είσοδο ένα όρισμα από τον χρήστη το οποίο αν είναι σε μορφή postfix είναι έγκυρη τη δέχεται και τη μετατρέπει σε μορφή infix, αν δε είναι έγκυρη τυπώνει μήνυμα λάθους. Η μετατροπή γίνεται χρησιμοποιώντας τη λίστα του μέρους α. Για να βρούμε την εν θεματική μορφή διατρέχουμε τη μεταθεματική μας μορφή τύπου String. Αν είναι αριθμός τον προσθέτουμε στο τέλος της λίστας, αν είναι τελεστής αφαιρούμε από το τέλος της λίστας τους 2 τελευταίους κόμβους, κάνουμε την πράξη με τον τελεστή και προσθέτουμε το αποτέλεσμα στο τέλος της

λίστας. Επαναλαμβάνουμε μέχρι να διατρέξουμε όλη την λίστα όπου έχει μείνει ένας κόμβος και η συνάρτηση μας επιστρέφει αυτή την τιμή με την μέθοδο removeLast() όπου είναι και το αποτέλεσμά μας.

c.

Στο πρόγραμμα πελάτη δεχόμαστε μια μεταβλητή σαν είσοδο από τον χρήστη. Αν η μεταβλητή δεν είναι τύπου DNA εμφανίζουμε μήνυμα λάθους. Αν η μεταβλητή είναι τύπου DNA τότε διατρέχουμε τη μεταβλήτη τύπου string που δώσαμε σαν είσοδο. Για κάθε χαρακτήρα A που συναντάμε εισάγουμε στην αρχή της λίστας τον χαρακτήρα T. Αντίστοιχα όταν συναντάμε τον χαρακτήρα T εισάγουμε στην αρχή της λίστας τον χαρακτήρα A, τον χαρακτήρα G εισάγουμε στην αρχή της λίστας τον χαρακτήρα C , τον χαρακτήρα C εισάγουμε στην αρχή της λίστας τον χαρακτήρα G. Στη συνέχεια διατρέχουμε ξανά τη μεταβλητή που εισάγαμε και εξετάζουμε κάθε στοιχείο αν είναι ίδιο με αυτό που βρίσκεται στη κορυφή της λίστας κάθε φορά καλόντας τη μέθοδο removeFirst() που επιστρέφει και διαγράφει το πρώτο στοιχείο της λίστας. Αν όλα τα στοιχεία κατά τον παραπάνω έλεγχο βρεθούν ίσα τότε το DNA είναι παλίνδρομο και εμφανίζει το αντίστοιχο μήνυμα. Σε αντίθετη περίπτωση θα εμφανίσει μήνυμα πως το DNA δεν είναι παλίνδρομο. Η πολυπλοκότητα του αλγορίθμου είναι O(2N) δηλαδή γενικά O(N) αφού διατρέχει δυο for loop N στοιχείων.