



ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΜΥΥ-105 / ΠΛΥ-106: ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ

4ο ΕΡΓΑΣΤΗΡΙΟ

(ΧΕΙΜΕΡΙΝΟ ΕΞΑΜΗΝΟ 2013-2014)

Διδάσκοντες

Α. Κόντης, Γ. Μανής

Υπεύθυνη Εργαστηρίου

Μαρία Χρόνη

Άσκηση 1. Παλινδρομικές Συμβολοσειρές

Μια συμβολοσειρά ονομάζεται *παλινδρομική* (ή *καρτινική*) αν διαβάζεται το ίδιο από αριστερά προς τα δεξιά και από τα δεξιά προς τα αριστερά. Για παράδειγμα τα ονόματα Άννα, Σάββας, Otto είναι παλινδρομικές συμβολοσειρές, όπως επίσης και οι λέξεις *civic*, *radar*, *level*, *rotor*, *noon*, *kayak*, *reviver*, *madam*. Η γνωστότερη παλινδρομική (καρτινική) επιγραφή είναι η: “Νίψον ανομήματα μη μόναν όψιν”. Για κάθε παλινδρομική συμβολοσειρά ισχύουν τα ακόλουθα:

- Μια συμβολοσειρά που περιέχει ένα ή κανένα χαρακτήρα είναι παλινδρομική.
- Αν η συμβολοσειρά αποτελείται από δύο ή περισσότερους χαρακτήρες είναι παλινδρομική εάν ο πρώτος και τελευταίος χαρακτήρας είναι ίσοι και η συμβολοσειρά ανάμεσα στο πρώτο και τελευταίο χαρακτήρα είναι επίσης παλινδρομική.

Με βάση τα παραπάνω να γράψετε ένα πρόγραμμα το οποίο:

α) Θα ζητείται από το χρήστη ως είσοδο μια συμβολοσειρά.

β) Θα δημιουργήσετε μια συνάρτηση `CheckPalindrome`, η οποία θα δέχεται ως όρισμα τη συμβολοσειρά που έδωσε ο χρήστης ως είσοδο και θα επιστρέφει `True` αν η συμβολοσειρά είναι παλινδρομική αλλιώς θα επιστρέφει `False`.

γ) Θα δημιουργήσετε μια αναδρομική συνάρτηση `RecCheckPalindrome`, η οποία θα δέχεται ως όρισμα τη συμβολοσειρά που έδωσε ο χρήστης ως είσοδο και θα ελέγχει αν μια συμβολοσειρά είναι παλινδρομική.

Το πρόγραμμά σας θα καλεί και τις δύο συναρτήσεις και θα τυπώνει κατάλληλα μηνύματα.

Αποθηκεύστε το αρχείο ως `palindrome.py`.

Άσκηση 2. Συμβολοσειρές

Μια ακολουθία *DNA* (δεσοξυριβονουκλεϊκό οξύ) αποτελείται από τις αζωτούχες βάσεις αδενίνη (A), γουανίνη (G), κυτοσίνη (C), και θυμίνη (T) ενώ μια ακολουθία *RNA* (ριβονουκλεϊκό οξύ) αποτελείται από τις αζωτούχες βάσεις αδενίνη (A), γουανίνη (G), κυτοσίνη (C), και ουρακίλη (U). Ένα παλίνδρομο σε μια ακολουθία *DNA* ή *RNA*, ονομάζεται *συμπληρωματικό παλίνδρομο* (*complemented palindrome*), αν προκύπτει από την αντικατάσταση όλων των χαρακτήρων της ακολουθίας, δηλαδή από την αρχή έως τη μέση της ακολουθίας, με τις αντίστοιχες συμπληρωματικές αζωτούχες βάσεις. Για το *DNA* οι αζωτούχες βάσεις A και C είναι συμπληρωματικές των T και G αντίστοιχα, ενώ για το *RNA* οι βάσεις A και C είναι συμπληρωματικές των U και G αντίστοιχα. Για παράδειγμα η συμβολοσειρά: $X = AGCTCGCGAGCT$ αποτελεί ένα συμπληρωματικό παλίνδρομο, αφού προκύπτει με την αντικατάσταση των χαρακτήρων $X[1 : 6]$ με τις συμπληρωματικές βάσεις τους που ισοδυναμούν με τις βάσεις στις θέσεις $X[7 : 12]$. Θα γράψετε ένα πρόγραμμα το οποίο θα ζητά από το χρήστη να εισάγει μια συμβολοσειρά αρτίου μήκους από αζωτούχες βάσεις και θα ελέγχει αν η συμβολοσειρά είναι *συμπληρωματικό παλίνδρομο*. Το πρόγραμμά σας θα αποτελείται από τις ακόλουθες συναρτήσεις:

- `is_valid(inputString)`, η οποία θα δέχεται ως όρισμα τη συμβολοσειρά που δίνει ο χρήστης ως είσοδο και θα επιστρέφει *True* εάν η συμβολοσειρά είναι έγκυρη, αν δηλαδή αποτελείται από τους χαρακτήρες (ή, ισοδύναμα από τις αζωτούχες βάσεις) A, T, C, G, U, αλλιώς θα επιστρέφει *False*.
- `check_DNA_RNA(str)`, η οποία θα δέχεται ως όρισμα μια έγκυρη συμβολοσειρά και θα επιστρέφει "DNA" ή "RNA", το είδος δηλαδή του νουκλεϊκού οξέος.
- `get_complement(str1, str2)`, η οποία θα δέχεται ως ορίσματα το πρώτο μισό της αρχικής συμβολοσειράς καθώς και το είδος της και θα επιστρέφει τη συμπληρωματική της.
- `check_complpalindrome(str1, str2)`, η οποία θα δέχεται ως ορίσματα τη συμπληρωματική συμβολοσειρά και την αρχική, θα ελέγχει αν η συμβολοσειρά είναι συμπληρωματικό παλίνδρομο και θα τυπώνει ανάλογο μήνυμα.

Αποθηκεύστε το αρχείο ως `complPalindrome.py`.

Άσκηση 3. Πολλαπλασιασμός δύο Ακεραίων (Αναδρομικά)

Να γράψετε ένα πρόγραμμα που θα ζητά από το χρήστη δύο ακέραιους αριθμούς (α, β), και θα υπολογίζει το γινόμενο τους. Για το σκοπό αυτό θα δημιουργήσετε μια αναδρομική συνάρτηση, την `recuMul(α, β)`, η οποία θα δέχεται ως ορίσματα τους δύο αυτούς αριθμούς και θα επιστρέφει το γινόμενο τους.

Αποθηκεύστε το αρχείο ως `RecMultiply.py`.