



ΕΡΓΑΣΤΗΡΙΟ 9

Άσκηση 1. Reversal of Elements

Συμπληρώστε το παρακάτω πρόγραμμα έτσι ώστε να αντιστρέφει τη σειρά των τιμών των δεδομένων, και στη συνέχεια να τα εκτυπώνει.

```
class ReverserVersion1
{
    public static void main ( String[] args )
    {
        int[] data = {11,25,43,4,15,96,78,18,9,30,71,42,13,64};

        // reverse the data
        for ( int j=0; j < data.length; j++)
        {
            // swap data[j] and data[data.length-j-1]
            int temp = data[j];
            data[j] = data[data.length-j-1];
            data[data.length-j-1] = temp;
        }

        // print the reversed data
        for ( int j=0; j < data.length; j++)
        {
            System.out.print(data[j] + " ");
        }
    }
}
```

Άσκηση 2. Smooth Operator

Ένα ηχητικό σήμα μερικές φορές αποθηκεύεται ως μια λίστα με ακέραιων τιμών. Οι τιμές αντιπροσωπεύουν την ένταση του σήματος σε διαδοχικά χρονικά διαστήματα. Φυσικά, σε ένα πρόγραμμα το σήμα θα εκπροσωπείται με έναν πίνακα ακέραιων.

Συχνά, μια μικρή ποσότητα θορύβου περιλαμβάνεται στο σήμα. Ο θόρυβος που είναι συνήθως μικρός, αντιπροσωπεύει προσωρινές αλλαγές στη στάθμη του σήματος

Η εξομάλυνση (smoothing) ενός σήματος αφαιρεί το θόρυβο και βελτιώνει την αντιληπτή ποιότητα του σήματος. Η άσκηση αυτή στοχεύει να εξομαλύνει τις τιμές σε έναν ακέραιο πίνακα που αντιπροσωπεύει ένα ηχητικό σήμα.

Υποθέστε ότι οι αρχικές τιμές είναι στον πίνακα "signal". Υπολογίστε τον νέο πίνακα smooth (που αντιπροσωπεύει το εξομαλυμένο πίνακα smoothed array) όπου:

κάθε τιμή $smooth[N]$ είναι ο μέσος όρος των τριών τιμών: $signal[N-1]$, $signal[N]$, και $signal[N+1]$.



Το πρώτο στοιχείο του πίνακα `smooth`, θεωρήστε ότι είναι ο μέσος όρος των δύο πρώτων στοιχείων του πίνακα `signal`. Για το τελευταίο στοιχείο του πίνακα `smooth`, θεωρήστε ότι είναι ο μέσος όρος των δύο τελευταίων στοιχείων του πίνακα `signal`.

Χρησιμοποιήστε αριθμητική ακεραίων, έτσι ώστε οι τιμές στον πίνακα `smooth` να είναι ακέραιοι.

```
public class Smooth {

    public static void main ( String[] args ){
        int[] signal = {1, 5, 4, 5, 7, 6, 8, 6, 5, 4, 5, 4};
        int[] smooth;

        // compute the smoothed value for each
        // cell of the array smooth
        smooth[0] =
        smooth[ signal.length-1 ] =
        for ( ) {

        }

        // write out the input
        for ( int j=0; j < smooth.length; j++) {

        }

        // write out the result
        for ( int j=0; j < smooth.length; j++) {

        }

    }
}
```

Για την ερμηνεία των αποτελεσμάτων, θυμηθείτε ότι η διαίρεση ακέραιων απορρίπτει το υπόλοιπο. Δεν υπολογίζει μια στρογγυλεμένη τιμή. Εδώ είναι ένα παράδειγμα εκτέλεσης του προγράμματος:

```
>java Smooth
signal: 1 5 4 5 7 6 8 6 5 4 5 4
smooth: 3 3 4 5 6 7 6 6 5 4 4 4
```

Άσκηση 3. Counting characters

Γράψετε ένα πρόγραμμα που διαβάζει κείμενο (ακολουθία από χαρακτήρες) από ένα txt αρχείο με ανακατεύθυνση και παρουσιάζει στην οθόνη πόσες φορές εμφανίζεται ο κάθε χαρακτήρας (a – z). Υποθέστε ότι το αρχείο εισόδου περιέχει μόνο χαρακτήρες του αγγλικού αλφαβήτου, αριθμούς, κόμματα, κενά και μόνο μια τελεία στο τέλος του αρχείου. Οι αριθμοί, τα κόμματα, τα κενά (spaces) και οι αλλαγές γραμμών να αγνοούνται. Το πρόγραμμα δεν θα πρέπει να κάνει



διάκριση κεφαλαίων και μικρών χαρακτήρων. Η ανάγνωση δεδομένων από το πρόγραμμά σας τερματίζεται όταν διαβαστεί η τελεία.

Παράδειγμα εκτέλεσης:

```
java countChars < data.txt
```

```
a -> 14
```

```
b -> 3
```

```
...
```