



ΑΛΓΟΡΙΘΜΟΙ

Διδάσκουσα: Κ. Παπακωνσταντινοπούλου

2η Σειρά Προγραμματιστικών και Γραπτών Ασκήσεων Ημ/νία παράδοσης: 14/4/2019

Οι λύσεις των προγραμματιστικών ασκήσεων πρέπει να υλοποιηθούν σε γλώσσα προγραμματισμού **Java** και τα **αρχεία** σας πρέπει να υποβληθούν στο **eClass** σε ένα συμπίεσμένο φάκελο .zip με όνομα τον αριθμό μητρώου σας, δηλαδή 3xxxxxx.zip.

Τις λύσεις των γραπτών ασκήσεων, καθώς και την αναδρομική εξίσωση και την πολυπλοκότητα που αντιστοιχεί σε κάθε αλγόριθμο των προγραμματιστικών ασκήσεων (θα δοθεί αναλυτική περιγραφή) πρέπει να τις γράψετε σε ηλεκτρονική μορφή και να τις συμπεριλάβετε σε ένα pdf αρχείο μέσα στον παραπάνω φάκελο.

Προγραμματιστικές Ασκήσεις

Άσκηση 2.1

Σε ένα ηλεκτρονικό παιχνίδι κεντρικός ήρωας είναι ένας βάτραχος, ο οποίος βρίσκεται σε μια λίμνη με νούφαρα και πρέπει να φτάσει από το φύλλο 1 (αφετηρία) στο φύλλο N (προορισμός), όσο γίνεται πιο γρήγορα. Μεταξύ των φύλλων 1 και N , σε ευθεία γραμμή, βρίσκονται άλλα $N - 2$ φύλλα.

Ο βάτραχος όντας πάνω σε ένα φύλλο μπορεί να κάνει άλμα και να προσγειώνεται σε κάποιο άλλο φύλλο. Υποθέτουμε ότι όλα τα άλματα έχουν την ίδια διάρκεια. Πάνω σε κάθε φύλλο υπάρχει μια ποσότητα φαγητού x που, αν καταναλωθεί, επιτρέπει στο βάτραχο να κάνει ένα άλμα προς ένα οποιοδήποτε φύλλο μέχρι και τα x επόμενα. Υποθέτουμε ότι ο βάτραχος τρώει την τροφή σε κάθε φύλλο στο οποίο φτάνει.

Για παράδειγμα, αν F είναι ο πίνακας με τις ποσότητες φαγητού σε κάθε φύλλο και $F[] = \{1, 4, 5, 6, 7, 3, 2, 5, 8, 10, 6, 8\}$, οπότε ο βάτραχος πρέπει να πάει από το φύλλο 1 (με φαγητό 1, στοιχείο $F[0]$) στο φύλλο 12 (με φαγητό 8, στοιχείο $F[11]$) τότε:

- Από το φύλλο 1 θα κάνει αρχικά ένα άλμα μέχρι το φύλλο 2.
- Η τροφή στο φύλλο 2 επαρκεί για άλμα μέχρι και 4 φύλλα μετά, οπότε στη συνέχεια μπορεί να πάει σε ένα από τα φύλλα:
 - 3 (που έχει φαγητό 5),
 - 4 (που έχει φαγητό 6),
 - 5 (που έχει φαγητό 7) και

- 6 (που έχει φαγητό 3).

Αν επιλέξει να πάει στο φύλλο 5, στη συνέχεια μπορεί να κάνει άλμα μέχρι το φύλλο 12 και να τερματίσει.

Ο βάτραχος πρέπει να φτάσει στο φύλλο-προορισμό σε όσο το δυνατό λιγότερο χρόνο. Στο παραπάνω παράδειγμα, ο ελάχιστος δυνατός χρόνος μετάβασης στον προορισμό (φύλλο 12) είναι 3.

Να γράψετε πρόγραμμα Java που παίρνει ως είσοδο ένα μονοδιάστατο πίνακα ακεραίων F (από αρχείο), κι επιστρέφει τη διάρκεια της συντομότερης διαδρομής του βάτραχου προς τον προορισμό, κι ένα πίνακα με τα φύλλα από τα οποία θα περάσει κατά τη διαδρομή αυτή, χρησιμοποιώντας *δυναμικό προγραμματισμό*. Για παράδειγμα, η έξοδος για το παραπάνω στιγμιότυπο πρέπει να είναι:

3, {1, 2, 5, 12}.

Άσκηση 2.2

Θέλετε να βοηθήσετε ένα διαιτολόγο να φτιάχνει μενού συγκεκριμένου πλήθους θερμίδων που να έχουν την ελάχιστη δυνατή περιεκτικότητα σε λιπαρά. Υπάρχουν διαθέσιμα N είδη τροφής, καθένα από τα οποία προσφέρει συγκεκριμένες θερμίδες και έχει ορισμένη περιεκτικότητα σε λιπαρά. Κάθε μενού μπορεί να περιλαμβάνει κάθε είδος τροφής *μία μόνο φορά*.

Να γράψετε πρόγραμμα Java που παίρνει ως είσοδο έναν ακέραιο C που εκφράζει το πλήθος θερμίδων του ζητούμενου μενού και ένα $2 \times N$ πίνακα ακεραίων F (από αρχείο) όπου:

$F[1, i]$ = το πλήθος θερμίδων του είδους i ,

$F[2, i]$ = η περιεκτικότητα σε λιπαρά του είδους i ,

κι επιστρέφει το χαμηλότερης περιεκτικότητας σε λιπαρά μενού που δίνει C θερμίδες (ή όσο γίνεται πιο κοντά στις C , χωρίς να τις υπερβαίνει), με τη βοήθεια του *δυναμικού προγραμματισμού*. Συγκεκριμένα, η έξοδος θα πρέπει να είναι της μορφής

$\tilde{C}, \{m_1, m_2, \dots, m_k\}$

όπου \tilde{C} είναι το πλήθος θερμίδων του μενού που επιλέχθηκε και $\{m_1, m_2, \dots, m_k\}$ ο πίνακας με τα είδη τροφής που περιλαμβάνει το επιλεγμένο μενού.

Βοήθεια (και για τις δύο ασκήσεις):

Παρέχονται (στην ενότητα ‘Εγγγραφα’ του eClass, στο φάκελο ‘Auxiliary source files’) συναρτήσεις για το διάβασμα ενός αρχείου που περιέχει ακεραίους χωρισμένους με ένα κενό μεταξύ τους ή διδιάστατου πίνακα ακεραίων χωρισμένων με ένα κενά μεταξύ τους και την αποθήκευσή τους σε μια λίστα ή ένα πίνακα αντίστοιχα.

Επίσης παρέχονται (στο φάκελο ‘Datasets and Test files’) ενδεικτικά αρχεία εισόδου που μπορείτε να χρησιμοποιήσετε για να δοκιμάζετε το πρόγραμμά σας. Προφανώς μπορείτε να δημιουργήσετε και να χρησιμοποιήσετε αντίστοιχα δικά σας αρχεία για δοκιμή των υλοποιήσεών σας, προσέχοντας όμως να διατηρήσετε το ίδιο format.

Οδηγίες υποβολής:

- Οι λύσεις των δύο ασκήσεων θα πρέπει να περιέχονται σε διαφορετικά αρχεία (1 class για κάθε άσκηση, πχ. Exercise1.java, Exercise2.java).
- Θα πρέπει να έχετε ένα τρίτο αρχείο για την εξέταση των 2 ασκήσεων (ExercisesSet2.java), με τη μέθοδο ‘main’.
- Τα αρχεία που θα χρησιμοποιήσετε για τις δοκιμές σας (πχ 2.1-sm.txt, 2.2-sm.txt), θα πρέπει να βρίσκονται στον ίδιο φάκελο με τα παραπάνω.
- Στην ενότητα ‘Εγγγραφα’, στο φάκελο ‘Submission templates’ μπορείτε να βρείτε ένα πρότυπο του πώς πρέπει να είναι ο συμπιεσμένος φάκελος που θα υποβάλετε.