ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Εαρινό 2018

ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ Ι

Άσκηση 2

Καταληκτική ημερομηνία και ώρα ηλεκτρονικής υποβολής: 20/5/2018, 23:59:59

Πίστες, κλειδιά και αστέρια (0.25+0.25 = 0.5 βαθμοί)

Παίζουμε ένα ηλεκτρονικό παιχνίδι που αποτελείται από N+1 πίστες, αριθμημένες από 0 έως N. Το παιχνίδι ξεκινάει υποχρεωτικά με την πίστα 0, που είναι εισαγωγική. Κάθε πίστα που ολοκληρώνουμε επιτυχώς μας επιβραβεύει με κάποια «αστέρια», ανάλογα με τη δυσκολία της, και επίσης με κάποια «κλειδιά», πολλών διαφορετικών σχημάτων και μεγεθών, τα οποία μπορούμε να χρησιμοποιήσουμε για να ανοίξουμε και να παίξουμε άλλες, επόμενες πίστες. Προσέξτε ότι δεν χρειάζεται (και ίσως δεν είναι καν δυνατόν) να παίξουμε τις επόμενες πίστες κατ' αριθμητική σειρά: για να ανοίξουμε και να παίξουμε μία πίστα πρέπει να διαθέτουμε το κατάλληλο σύνολο από κλειδιά. Επιπλέον, τα κλειδιά είναι μίας χρήσης: μόλις χρησιμοποιηθεί ένα κλειδί καταστρέφεται και δεν μπορεί να ξαναχρησιμοποιηθεί. Αφού λοιπόν παίξουμε την εισαγωγική πίστα, έχουμε δικαίωμα να επιλέξουμε ποια άλλη πίστα θα ανοίξουμε και θα παίξουμε, αρκεί φυσικά να μην την έχουμε ξαναπαίξει και να έχουμε στη διάθεσή μας όλα τα κλειδιά που χρειάζονται. Το παιχνίδι τελειώνει όταν δεν μπορούμε να παίξουμε καμία άλλη πίστα, είτε γιατί δεν έχει μείνει καμία, είτε γιατί δεν έχουμε διαθέσιμα κλειδιά για να ανοίξουμε όσες έχουν μείνει.

Αυτό που ζητάει η άσκηση είναι να γραφούν δύο προγράμματα (ένα σε ML και ένα σε Java) τα οποία να παίρνουν ως είσοδο τις περιγραφές των πιστών και να εκτυπώνουν το μέγιστο δυνατό πλήθος αστεριών που μπορούμε να έχουμε κερδίσει στο τέλος του παιχνιδιού.

Τα στοιχεία εισόδου θα διαβάζονται από ένα αρχείο με μορφή σαν και αυτή που φαίνεται στα παραδείγματα παρακάτω. Η πρώτη γραμμή του αρχείου έχει τον αριθμό **N**. Οι επόμενες N+1 γραμμές περιέχουν τις περιγραφές των πιστών, ξεκινώντας από την 0 και προχωρώντας μέχρι αυτήν που έχει αύξοντα αριθμό N. Κάθε τέτοια γραμμή αρχίζει με τρεις αριθμούς $\mathbf{k}_i$, $\mathbf{r}_i$ και $\mathbf{s}_i$ — το πλήθος των κλειδιών που απαιτούνται για το άνοιγμά της, το πλήθος των κλειδιών που αποκτούμε όταν την ολοκληρώσουμε, και το πλήθος των αστεριών που κερδίζουμε, αντίστοιχα — και συνεχίζουν με τους κωδικούς αριθμούς $\mathbf{k}_i + \mathbf{r}_i$ κλειδιών. Θεωρήστε ότι τα κλειδιά είναι αριθμημένα με φυσικούς αριθμούς από 1 έως 99.999. Επίσης, θεωρήστε δεδομένο ότι η εισαγωγική πίστα δεν θα απαιτεί κανένα κλειδί (δηλαδή ότι το $\mathbf{k}_i$ της θα είναι μηδέν.

Περιορισμοί: $1 \le \mathbf{N} \le 42$, $0 \le \mathbf{k}_i \le 10$, $0 \le \mathbf{r}_i \le 10$, $0 < \mathbf{s}_i$, $\Sigma\mathbf{s}_i \le 10^9$. Όριο μνήμης: 4GB, όριο χρόνου εκτέλεσης: 1 λεπτό.

Παρακάτω δίνονται κάποια παραδείγματα σε ML και σε Java.

Σε MLton ή σε OCaml

```
$ ./pistes p1.txt
610

$ ./pistes p2.txt
440
```

Σε SML/NJ

```
- pistes "p1.txt";
610
val it = () : unit
- pistes "p2.txt";
440
val it = () : unit
```

Σε Java

```
- java Pistes p1.txt
610
- java Pistes p2.txt
440
```

όπου τα αρχεία με τα δεδομένα εισόδου είναι τα εξής (η εντολή `cat` είναι εντολή του Unix):

```
$ cat p1.txt
4
0 1 0 1
1 0 200 1
1 2 150 1 1 3
2 0 120 2 1
1 2 140 3 1 2
```

```
$ cat p2.txt
5
0 3 10 1 3 1
2 1 100 1 3 2
3 2 900 1 1 2 5 3
2 2 200 5 1 1 3
1 2 130 2 3 5
2 2 120 3 2 1 1
```

Στο πρώτο παράδειγμα, μπορεί κανείς να παίξει την εισαγωγική πίστα από την οποία αποκτά το κλειδί 1 αλλά κανένα αστέρι. Στη συνέχεια, μπορεί να χρησιμοποιήσει το κλειδί 1 για να ανοίξει και να παίξει την πίστα 2. Όταν την ολοκληρώσει, θα έχει 150 αστέρια και τα κλειδιά 1 και 3. Μετά, μπορεί να χρησιμοποιήσει το κλειδί 3 για να ανοίξει και να παίξει την πίστα 4, που θα του δώσει 140 αστέρια και τα κλειδιά 1 και 2 (άρα θα έχει τώρα 290 αστέρια και τα κλειδιά 1, 1 και 2 — δηλαδή δύο αντίτυπα του κλειδιού 1 και ένα αντίτυπο του κλειδιού 1). Έπειτα, μπορεί να χρησιμοποιήσει τα κλειδιά 2 και 1 (τό ένα από τα δύο αντίτυπα που διαθέτει) για να ανοίξει και να παίξει την πίστα 3, που θα του δώσει 120 αστέρια και κανένα επιπλέον κλειδί. Τέλος, μπορεί να χρησιμοποιήσει το κλειδί 1 (το δεύτερο αντίτυπο) για να ανοίξει και να παίξει την πίστα 1, που θα του δώσει 200 αστέρια και κανένα επιπλέον κλειδί. Το παιχνίδι τελειώνει γιατί δεν υπάρχει άλλη πίστα που να μην έχει ανοίξει. Ο παίκτης έτσι θα έχει πάρει 150+140+120+200 = 610 αστέρια. Αυτό είναι προφανώς και το μέγιστο εφικτό για αυτό το παράδειγμα.

Η συντέλεια του κόσμου, ξανά (0.25+0.25 = 0.5 βαθμοί)

Το πρόβλημα με τη συντέλεια του κόσμου είναι γνωστό από την πρώτη σειρά ασκήσεων της φετινής χρονιάς. Το ζητούμενο αυτής της άσκησης είναι να γραφεί η λύση του σε δύο ακόμη γλώσσες: Java και Python 3. (Επειδή οι υλοποιήσεις της Python δεν τρέχουν native code, ο χρονικός περιορισμός για αυτή την άσκηση σε αυτή τη γλώσσα θα είναι σημαντικά αυξημένος.) Το πρόγραμμά σας θα πρέπει να έχει την ίδια συμπεριφορά με τα προγράμματα σε C/C++ που παραδώσατε για την πρώτη σειρά ασκήσεων. Για τα παραδείγματα της εκφώνησης της πρώτης σειράς, η έξοδος του προγράμματός σας πρέπει να είναι η ακόλουθη:

Σε Java

```
$ java Doomsday d1.txt
4
++++*----
```

```
$ java Doomsday d2.txt
5
++++**----
```

```
$ java Doomsday d3.txt
the world is saved
+++X..X---
++++XX----
```

```
$ java Doomsday d4.txt
5
+X..XX-----
+X..X--X---
+X..-------
+++XX-*----
XXX++X**---
--X++++*XXX
---XXX++X--
------*X---
```

Σε Python

```
$ python3 doomsday.py d1.txt
4
++++*----
```

```
$ python3 doomsday.py d2.txt
5
++++**----
```

```
$ python3 doomsday.py d3.txt
the world is saved
+++X..X---
++++XX----
```

```
$ python3 doomsday.py d4.txt
5
+X..XX-----
+X..X--X---
+X..-------
+++XX-*----
XXX++X**---
--X++++*XXX
---XXX++X--
------*X---
```

Περαιτέρω οδηγίες για τις ασκήσεις

- Μπορείτε να δουλέψετε σε ομάδες το πολύ δύο ατόμων. Μπορείτε αν θέλετε να σχηματίσετε διαφορετική ομάδα σε σχέση με την προηγούμενη σειρά ασκήσεων – οι ομάδες στο σύστημα υποβολής είναι έτσι και αλλιώς καινούργιες για κάθε σειρά ασκήσεων.

- Δεν επιτρέπεται να μοιράζεστε τα προγράμματά σας με συμφοιτητές εκτός της ομάδας σας ή να τα βάλετε σε μέρος που άλλοι μπορούν να τα βρουν (π.χ. σε κάποια σελίδα στο διαδίκτυο, σε ιστοσελίδες συζητήσεων, …). Σε περίπτωση που παρατηρηθούν «περίεργες» ομοιότητες σε προγράμματα, ο βαθμός των εμπλεκόμενων φοιτητών σε *όλες τις σειρές ασκήσεων* γίνεται αυτόματα μηδέν ανεξάρτητα από το ποια ομάδα... «εμπνεύστηκε» από την άλλη.

- Μπορείτε να χρησιμοποιήσετε «βοηθητικό» κώδικα (π.χ. κάποιο κώδικα που διαχειρίζεται κάποια δομή δεδομένων) που βρήκατε στο διαδίκτυο στα προγράμματά σας, με την προϋπόθεση ότι το πρόγραμμά σας περιέχει σε σχόλια την παραδοχή για την προέλευση αυτού του κώδικα και ένα σύνδεσμο σε αυτόν.

- Τα προγράμματα σε ML πρέπει να είναι σε ένα αρχείο και να δουλεύουν σε SML/NJ v110.76 ή σε MLton 20100608 ή σε Objective Caml version 4.01.0. Το σύστημα ηλεκτρονικής υποβολής επιτρέπει να επιλέξετε μεταξύ αυτών των υλοποιήσεων της ML.

- Ο κώδικας των προγραμμάτων σε Java μπορεί να βρίσκεται σε περισσότερα του ενός αρχείου αν θέλετε, αλλά θα πρέπει να μπορεί να μεταγλωττιστεί χωρίς προβλήματα με τον Java compiler με εντολές της μορφής: **`javac Pistes.java`** και **`javac Doomsday.java`**. Μην ορίσετε δικά σας packages! Η υποβολή σας σε Java μπορεί είτε να είναι ένα μόνο **`.java`** αρχείο ή να αποτελείται από ένα **`.zip`** αρχείο ενός directory το οποίο περιέχει τα **`.java`** αρχεία της υποβολής σας (και μόνο αυτά – μην υποβάλετε **`.class`** αρχεία). Σε κάθε περίπτωση, η υποβολή σας πρέπει να έχει ένα αρχείο με τα ονόματα που φαίνονται παραπάνω σε αυτήν την παράγραφο.

- Τα προγράμματα σε Python πρέπει να είναι σε ένα αρχείο και να δουλεύουν σε Python 3.4.2. (Προσέξτε ότι η Python 2 είναι διαφορετική διάλεκτος της Python!)

- Η υποβολή των προγραμμάτων θα γίνει ηλεκτρονικά μέσω του moodle, όπως και στην προηγούμενη άσκηση, και για να μπορέσετε να τις υποβάλλετε, τα μέλη της ομάδας σας (και οι δύο) θα πρέπει να έχουν ήδη λογαριασμό στο moodle. Θα υπάρξει σχετική ανακοίνωση μόλις το σύστημα υποβολής καταστεί ενεργό. Τα προγράμματά σας πρέπει να διαβάζουν την είσοδο όπως αναφέρεται και δεν πρέπει να έχουν κάποιου άλλους είδους έξοδο εκτός από τη ζητούμενη διότι δε θα γίνουν δεκτά από το σύστημα υποβολής.