Eidgenössische Technische Hochschule Zürich
Ecole polytechnique fédérale de Zurich
Politecnico federale di Zurigo
Federal Institute of Technology at Zurich

Department Informatik
Markus Püschel David Steurer Peter Widmayer
Chih-Hung Liu
Stefano Leucci

23. Oktober 2017

Datenstrukturen & Algorithmen Blatt P5 HS 17

Hand-in: Bis Sonntag, 5. November 2017, 23:59 Uhr via Online Judge (nur Source Code). Fragen zur Aufgabenstellung oder Übersetzung werden wie üblich im Forum beantwortet.

Exercise P5.1 *Sliding token.*

Consider a n-vertex directed acyclic graph $G = (V, E)$ where $V = \{0, 1, \dots, n-1\}$. A coin is initially placed on vertex 0 and two players, Alice and Bob, take turns sliding it along the edges of G. More precisely, a turn consists of removing the token from its current vertex u and placing it on another vertex v such that $(u, v) \in E$. The first player that can no longer move the coin loses (and the other player wins). Alice goes first.

Your task is to design an algorithm that, given G, determines whether it is possible for Alice to always win the game (regardless of Bob's moves).

Input The input consists of a set of instances, or *test-cases*, of the previous problem. The first line of the input contains the number T of test-cases. The first line of each test-case is the integer n. The next n lines each describe a vertex and its outgoing edges in G. In particular, the $(i+2)$-th line of the test-case contains a list of integers $d, v_1, v_2, \dots, v_d$ where d is the *outdegree* of the i-th vertex in G and $(i, v_j) \in E\ \forall j = 1, \dots, d$.

Output The output consists of T lines, each containing a single character. The i-th line is the answer to the i-th test-case and is either "`A`" if Alice can win the game, or "`B`" otherwise (i.e., if Bob can always win).

Grading You get 3 bonus points if your program works for all inputs. Your algorithm should have an asymptotic time complexity of $O(|V| + |E|)$ with reasonable hidden constants. Submit your `Main.java` at `https://judge.inf.ethz.ch/team/websubmit.php?cid=18997&problem=DA17P4.1`. The enrollment password is "`asymptotic`".

Example

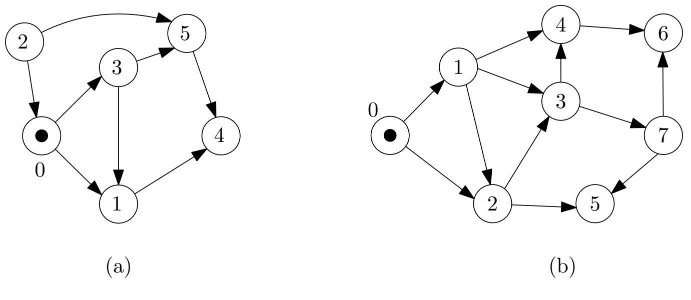

Figure 1: (a) A graph G in which Alice can always win. (b) A graph G in which Bob can always win.

Input (corresponding to the graphs in Figure 1):

```
2
6
2 1 3
1 4
2 5 0
2 5 1
0
1 4
8
2 1 2
3 2 3 4
2 3 5
2 7 4
1 6
0
0
2 5 6
```

Output:

```
A
B
```

Notes For this exercise we provide an archive on the lecture website containing a program template that will load the input and write the output for you. The archive also contains additional test cases (which differ from the ones used for grading). Your may import the class `java.util.ArrayDeque`. Importing other classes is **not allowed** (with the exception of the already imported `java.util.Scanner` class).