

BLG336E - Analysis of Algorithms II

2018–2019 Spring, Project 3

30.04.2019

Submission Deadline: **14.05.2019 23:59**

Overview

In this project, you are asked to implement a maximum flow algorithm in order to solve the given problem.

Soccer League

In this problem, you need to determine whether a given team has a chance to win the league or not.

- There are N different teams in the league, where $2 \leq N \leq 10$.
- **There is always a winner in each match, no draws.**
- **Teams do not play their matches in a fixed order**, teamA may have completed all its matches whereas teamB still has to play 4 more matches to play.
- Each win grants 1 point to the winning team and losing team gets nothing.
- If multiple teams have the same and highest score at the end of the league, all those teams win.

Input-Output Format

First line contains N , number of teams in the league. Second line contains points of each N teams in the league so far. Next N lines represent a symmetric $N \times N$ matrix showing remaining matches of the teams. Numbers in the same line are space-separated.

An example problem input and the answer can be seen below:

Input	Output
5 1 1 0 1 1 0 1 0 0 0 1 0 0 1 0 0 0 0 0 0 0 1 0 0 1 0 0 0 1 0	1 1 0 1 1

Input explanation:

$N=5$, number of teams in the league

Current scores of the teams in order: 1, 1, 0, 1, 1

It is seen that 3 more matches need to be played in the league.

1- Team1 vs Team2

2- Team2 vs Team4

3- Team4 vs Team5

Output explanation:

Each team except Team3 can still win the league.

A) A Simpler Elimination Rule (10 points)

Let the current team with the highest points in the league T, and its points P.

Instead of using a maximum-flow based method, could not we just use a simple criterion such that “If current points of a team plus the number of remaining matches a team has \geq P, that team still has a chance to win the league”.

Your answer must be yes or no. If your answer is yes, discuss/prove why. If your answer is no, discuss/prove why or give a counterexample scenario.

B) Implementation (50 points)

Implement a maximum flow based algorithm to decide whether given teams have a chance to win the league or not.

In your implementation, follow the given input-output format (especially output). Programs with different output format may result in a zero grade.

Input file name should be chosen by a command line argument. Sample input files are given. When the solution is found, if there is a second command line argument specified write your result to the specified output file, if there is not simply print your solution.

All your code must be written in C++, and should be compiled and run on ITU's Linux Server (you can access it through SSH) using g++. Otherwise your code will not be evaluated.

Your program should compile and run using the following commands:

You must provide instructions for how to compile your program with g++ (either in the report or in one of the source code files), and your program should be able to run the following commands:

```
g++ ...SOURCECODEFILES... -o project3  
  
./project3 <input-file-name> <output-file-name(OPTIONAL)>  
  
./project3 input1.txt  
  
./project3 input1.txt output1.txt
```

C) Report (40 points)

- Formulate the problem as a Network Flow problem. Visualize your network by stating flow values (15 points).
- Describe the method that you implemented for the league elimination in detail (15 points).
- Discuss the time and space complexities of your algorithm with their proofs (10 points).

Another example problem input-output below:

Input	Output
4 2 3 1 1 0 4 0 1 4 0 6 3 0 6 0 3 1 3 3 0	1 1 1 1

Note: If you have any questions, please feel free to contact Res. Asst. Türker Ünlü via e-mail (unlut@itu.edu.tr).

Policy:

- You may discuss the problem addressed by the project at an abstract level with your classmates, but you should not share or copy code from your classmates or from the Internet. You should submit your own, individual project.
- **Academic dishonesty including but not limited to cheating, plagiarism, collaboration is unacceptable and subject to disciplinary actions.**

Submission Instructions:

- Please submit your homework through Ninova e-Learning System.
- You must submit **all your source code file(s)** and **a softcopy report (PDF)**.
- All your code must be written in C++, and should be compiled and run on ITU's Linux Server (you can access it through SSH) using g++. Your code must compile without any errors; otherwise, you may get a grade of zero on the assignment.
- When you write your code, try to follow an object-oriented methodology with well-chosen variable, method, and class names and comments where necessary.
- You should be aware that the Ninova e-Learning System clock may not be synchronized with your computer, watch, or cell phone. Do not e-mail the teaching assistant or the instructors your submission after the Ninova site submission has closed. If you have submitted to Ninova once and want to make any changes to your report, you should do it before the Ninova submission system closes. Your changes will not be accepted by e-mail. Connectivity problems to the Internet or to Ninova in the last few minutes are not valid excuses for being unable to submit. You should not risk leaving your submission to the last few minutes. After uploading to Ninova, check to make sure that your project appears there.