CSCI-SHU 220: Algorithms @ NYU Shanghai — Last Updated: November 23, 2020
Homework 5 — Due Date: December 7, 2020, 11:00pm

---

You are allowed to discuss with others but not allow to use references other than the course notes and reference books. Please list your collaborators for each questions. Write your own solutions and make sure you understand them.

There are 55 marks in total. The full mark of this homework is 50. Your submission should be in PDF format generated by LaTeX. You may use the LaTeX template at:

https://www.overleaf.com/read/tsxxqdgjzhxx

Enjoy :).

## Problem 1: An important edge [10 marks]

An important edge in a connected undirected graph $G = (V, E)$ is an edge that, if removed, would disconnect $G$. Give an $O(|V| + |E|)$ algorithm to determine whether $G$ has an important edge.

## Problem 2: Unique Shortest Path [10 marks]

Given an undirected graph $G = (V, E)$ and two vertices $s, t \in V$, design a linear time algorithm to determine if there is a unique shortest path from $s$ to $t$.

## Problem 3: Reachability [10 marks]

Given an undirected graph $G = (V, E)$ design a linear time algorithm to find a vertex $s \in V$ from which all other vertices are reachable (i.e. there is a directed path from $s$ to $v$ for all $v \in V$) or report that none exists.

## Problem 4: Many stars [10 marks]

The following algorithm is performed on a connected undirected graph $G = (V, E)$, implemented by adjacency lists.

```
void mystery(v) {
    color v blue
    For each neighbor u of v
        if u is blue
            print *
```

```
        if u is red then
            Mystery(u)
}
int main() {
    color all vertices red
    Select an arbitrary vertex u
    mystery(u)
}
```

How many starts will be printed along the execution of `main`? Briefly explain your answer.

## Problem 5: Odd Cycle [10+5 marks]

Given a strongly connected directed graph $G = (V, E)$, design a linear time algorithm to determine if there is a directed odd cycle in $G$ or not.

**Bonus: (5 marks)** Prove the correctness of your algorithm.