PSG COLLEGE OF TECHNOLOGY

DEPARTMENT OF APPLIED MATHEMATICS AND COMPUTATIONAL SCIENCES

M.Sc (SS) – DESIGN AND ANALYSIS OF ALGORITHMS LAB

PROBLEM SHEET- VIII (Graphs)

1. Construct a undirected graph and perform the following operations on it:
   (i) Find the vertices adjacent to a given vertex
   (ii) Find vertices of a particular degree
   (iii) Find the sum of degree of all vertices
   (iv) Find whether a vertex has a path to another vertex
   (v) Depth first traversal
   (vi) Breadth first traversal
   (vii) Minimum spanning tree of a graph
   (viii) Shortest path from a particular vertex to all other vertices in the graph

2. You are given a string that contains only three characters a, b, and c. You can change at the most characters in the string. The uniformity index of a string is defined by the maximum length of the substring that contains the same character. Your task is to determine the maximum uniformity index than can be achieved.

Input format

- Two integers n and k where n denotes the size of the string and k as specified in the question
- A string of length

Output format
Print a single integer denoting the maximum uniformity that can be achieved

3. Our Code Monk recently learnt about Graphs and is very excited! He went over to the Graph-making factory to watch some freshly prepared graphs. Incidentally, one of the workers at the factory was ill today, so Monk decided to step in and do her job. The Monk's Job is to Identify whether the incoming graph is a tree or not. He is given **N**, the number of vertices in the graph and the **degree** of each vertex. Find if the graph is a tree or not.

Input:
First line contains an integer **N**, the number of vertices.
Second line contains **N** space-separated integers, the degrees of the **N** vertices.

Output:
Print "Yes" (without the quotes) if the graph is a tree or "No" (without the quotes) otherwise.

4. You are given flights route map of a country consisting of *N* cities and *M* undirected flight routes. Each city has an airport and each airport can work as layover. The airport will be in two states, Loading and Running. In loading state, luggage is loaded into the planes. In the running state, planes will leave the airport for the next city. All the airports will switch their states from Loading to Running and vice versa after every *T* minutes. You can cross a city if its airport state is running. Initially, all the airports are in running state. At an airport, if its state is loading, you have to wait for it to switch its state to running. The time taken to travel through any flight route is *C* minutes. Find the lexicographically smallest path which will take the minimum amount of time (in minutes) required to move from city *X* to city *Y*. It is guaranteed that the given flight route map will be connected. Graph won't contain multiple edges and self loops. A self loop is an edge that connects a vertex to itself.

   Input Format:

   The first line contains *4* space separated integers, *N*, *M*, *T* and *C*. Next *M* lines contains two space separated integers each, *U* and *V* denoting that there is a bidirectional road between city *U* and city *V*. Next line contains two space separated integers, *X* and *Y*.

   Output Format:

   In the first line print an integer *K*, denoting the number of city you need to go through to reach city *Y* from the city *X*. In next line, print *K* space separated integers denoting the path which will take the minimum amount of time (in minutes) required by to move from city *X* to city *Y*. There can be multiple paths. Print the lexicographically smallest one.

5. Leonard has decided to quit living with Dr. Sheldon Cooper and has started to live with Penny. Yes, you read it right. (And you read it here for the first time!) He is fed up of Sheldon, after all. Since, Sheldon no more has Leonard to drive him all around the city for various things, he's feeling a lot uneasy so he decides to set up a network of drivers all around the city to drive him to various places.

   But, not every driver wants to go every place in the city for various personal reasons, so Sheldon needs to trust many different cab drivers. (Which is a very serious issue for him, by the way!) The problem occurs mainly when Sheldon needs to go to - for example, the Comic book store - and there's no cab driver who goes directly to that place. So, he has to take a cab till another place, and then take a cab from there - making him more scared!

   Sheldon wants to limit his trust issues. Really. Once. And. For. All.

Let's say that you're given the schedule of all the cabs from the major points where he travels to and from - can you help Sheldon figure out the least number of cab drivers he needs to trust, in order to go to all the places he wants to?