ShanghaiTech University

CS240 Algorithm Design and Analysis

Spring 2021

Course Project

Due: 23:59, June 13, 2021

1. This project requires you to solve four problems. For each problem, write a program with the input and output formats as specified in the problem.
2. All programs must terminate within 10 seconds and use at most 4GB of memory. Programs exceeding either limit will not be accepted. You can write your programs in C, C++, Java or Python, though certain languages may lead to more efficient implementations.
3. You may not use third-party libraries in your solution, *e.g.* numpy in Python is not allowed. If you are not sure whether a library is third-party or not, you can try to submit your program, and it will cause a Runtime Error or Compile Error if it is.
4. Your program submission is scored by an Online Judge platform containing a number of test cases. Your score depends on the percentage of test cases your program passes.
5. The Online Judge website is `http://10.19.127.47`. You can also find the problem list at `http://10.19.127.47/contest/1/problems`.
6. Each student's account name and initial password are the **STUDENT ID of your gradescope account**. After logging into your account for the first time, please change your password. You can view your **STUDENT ID** at `https://www.gradescope.com/account/edit`. If you do not set it, your account name and initial password will be the prefix of your email in gradescope.
7. To avoid abuse of the OJ, you are allowed at most one submission every five minutes. Violations of this rule may result in penalties to your score.
8. You must **NOT**
   (a) Read or use solutions written by others.

(b) Allow other people to read or use your solutions.

(c) Obtain test data by repeated submissions or other means.

Problem 1:

Consider N balls located on the number line, with ball i located at coordinate X_i. Each ball i has a color C_i represented by an integer between 1 to N (inclusive). You begin at coordinate 0, and must collect all the balls by moving along the number line at a speed of 1 unit per second, and finally return to coordinate 0. You need to collect the balls in order of their colors, *i.e.* collect all the balls of color 1, then all the balls of color 2, etc. When collecting a ball, you have to be at the coordinate of the ball. However, when you are at a coordinate, you are not required to collect the ball there. Find the minimum time to collect all the balls, starting and ending at coordinate 0.

Constraints:

- $1 \leq N \leq 2 \times 10^5$
- $|X_i| \leq 10^9$
- $X_i \neq X_j$ $(i \neq j)$
- $X_i \neq 0$
- $1 \leq C_i \leq N$
- All values in the input are integers

Input:
Input is given from Standard Input in the following format:
N
X_1 C_1
X_2 C_2
X_3 C_3
$\cdots$
X_N C_N

Output:
Print the number of seconds needed.

Sample Input 1:
5
2 2
3 1
1 3
4 2
5 3

Sample Output 1:
12

The optimal strategy is:

- spend 3 seconds to reach coordinate 3 and collect ball 2 with color 1
- spend 1 second to reach coordinate 2 and collect ball 1 with color 2
- spend 2 seconds to reach coordinate 4 and collect ball 4 with color 2
- spend 1 second to reach coordinate 5 and collect ball 5 with color 3
- spend 4 seconds to reach coordinate 1 and collect ball 3 with color 3
- spend 1 second to return to coordinate 0

Sample Input 2:

```
9
5 5
-4 4
4 3
6 3
-5 5
-3 2
2 2
3 3
1 4
```

Sample Output 2:

```
38
```

Problem 2:

Consider a city with n water tanks numbered from 1 to n. These tanks are connected by unidirectional pipes in some manner, and any pair of water tanks is connected by at most one pipe in each direction. Each pipe has a strictly positive integer *width*, representing the amount of water the pipe can transport every unit of time. The water enters the city from the main water tank (numbered 1), and must go through some set of pipes to get to the sewer tank (numbered n).

You now want to increase the widths of a subset of the pipes by a total of $k > 0$ units, so that the amount of water which can be transmitted from the main water tank to the sewage tank is maximized. Given k, find this maximum amount of water.

Input:
The first line contains two space-separated integers n and k ($2 \leq n \leq 50, 0 \leq k \leq 1000$). On the next n lines, each line contains n integers separated by a single space. The $i+1$'st row and j'th column contains a number c_{ij}, equal to the width of the pipe from tank i to j ($0 \leq c_{ij} \leq 10^6, c_{ii} = 0$). If $c_{ij} = 0$, then there is no pipe from i to j.

Output:
Print a single integer – the maximum amount of water which can be transmitted from the main tank to the sewer tank per unit of time.

Sample Input:
```
5 7
0 1 0 2 0
0 0 4 10 0
0 0 0 0 5
0 0 0 0 10
0 0 0 0 0
```

Sample Output:
```
10
```

Problem 3:

Suppose you are given an undirected graph with N vertices and M edges. Determine whether it is possible to divide the vertices into non-empty sets $V_1, \cdots, V_k$ such that the following condition is satisfied:

- Every edge connects two vertices belonging to two "adjacent" sets. More formally, for every edge (i, j), there exists $1 \leq t \leq k - 1$, such that $i \in V_t, j \in V_{t+1}$ or $i \in V_{t+1}, j \in V_t$ holds.

If the answer is yes, find the maximum possible number of sets, k, in such a division.

Constraints:

- $2 \leq N \leq 200$
- N is an integer
- The given graph has no self loops
- The given graph is connected

Input:
The first line contains the number of vertices N and the number of edges M. Following this are M lines, each consisting of two number i and j, where $1 \leq i, j \leq N$ represent an undirected edge (i, j).

Output:
If it is impossible to divide the vertices into sets so that the condition is satisfied, print -1. Otherwise, print the maximum possible number of sets k in a division which satisfies the condition.

Sample Input 1:
2 1
1 2

Sample Output 1:
2

Notes 1:
We can put node 1 in V_1 and node 2 V_2.
Sample Input 2:
3 3
1 2
1 3
2 3

Sample Output 2:
-1

Problem 4:

Given an input $N > 0$, output an array A consisting of the integers $1, 2, \ldots, N$ in some order, with the property that for every $i < j$, there is no k, for $i < k < j$, such that $A[k] \times 2 = A[i] + A[j]$.

Input:
An integer $1 \leq N \leq 1000$.

Output:
Any array A with the above property.

Sample Input 1:
4

Sample Output 1:
$[2, 1, 4, 3]$

Sample Input 2:
5

Sample Output 2:
$[3, 1, 2, 5, 4]$