CS-2580: Hw3 Green Zone

Final Due Date: Sunday, March 6, 11:59pm

1 Problem Statement

The UN has been asked by an nation in turmoil to setup a green zone in the capital city to ensure peaceful international relations. However, global donations are running thin and the UN must use their precious resources efficiently. As head of UN logistics support it is your task to find the cheapest way to setup and deploy peace keeping forces throughout the city.

The input data for the Green Zone problem consists of \( n \) stations and \( m \) checkpoints. Each station, \( s \), has two properties, \( \text{cap}(s) \) the maximum number of peacekeepers that can work at \( s \), and \( \text{sc}(s) \) the cost of using \( s \). Each checkpoint, \( p \), has two properties, \( d(p) \) the amount of peacekeepers needed to ensure peace at \( p \), and \( \text{pc}(s,p) \) the cost of deploying \( d(p) \) peacekeepers from station \( s \) to checkpoint \( p \) (you can think of \( \text{pc}(s,p) \) as the cost of traveling from \( s \) to \( p \)). The optimization problem is to supply every checkpoint with the required peacekeepers while minimizing the costs of opening stations and deploying peacekeepers. Specifically, let \( W(s,p) \) be the number of peacekeepers from station \( s \) that goto checkpoint \( p \), then the objective is to minimize,

\[
\sum_s (\text{sc}(s) \times \sqrt{\text{W}(s,p) > 0}) + \sum_s \sum_p \frac{\text{pc}(s,p)}{d(p)} \times \text{W}(s,p).
\]

Additionaly, the amount of peacekeepers assigned to station, \( s \), cannot exceed the station capacity, \( \text{cap}(s) \) and all checkpoints must have at least \( d(p) \) peacekeepers assigned to them.

2 Assignment

Write a algorithm to solve the Green Zone problem, you can apply any technique you prefer, including but not limited to, LS, CP, IP, LP, DP, brute force etc. Your algorithm should be able to perform on all of the data sets in the course directory (including green zones with as many as 100 stations). The assignment should be performed by teams of at most 2 people. We always expect

- both source files and binary programs, if any, of the working algorithm;
- a specification on how to run the program;
- a brief report in plain text containing the names of each team member, a brief discussion of your solution strategy, implementation techniques and experimental observations. The report should be concise. An example report can be found here,

/course/cs258/data/README_example
3 I/O Specification

Format  The first line contains two numbers, $n$ followed by $m$. It is followed by $n$ lines, where each line $i$ represents $\text{cap}(i)$ followed by $\text{sc}(i)$. The end of this list is followed by $m$ collections of values. For each collection, $i$, the first number contains the value of $d(i)$ and following $n$ values representing $pc(s,i)$ (note that this list of $n$ values may span several lines).

[Input Format]
n m
cap_0 sc_0
cap_1 sc_1
...
d_0
pc_0_0 pc_1_0 pc_2_0 ... pc_n-1_0
d_1
pc_0_1 pc_1_1 pc_2_1 ... pc_n-1_1
...

The output has $n + 1$ lines. The first line contains two numbers: the first number is the best objective value found by the algorithm (a float); the second number is a flag, 1 if the algorithm is able to prove the optimality, otherwise 0. The following $n$ lines describe how the workers are allocated to the checkpoints. Each line, $i$ is a space separated list of $m$ values each representing the amount of workers from station $i$ that are deployed to checkpoint $j$.

[Output Format]
obj opt_flag
W(0,1) W(0,2) ... W(0,m-1)
...
W(n-1,1) W(n-1,2) ... W(n-1,m-1)

Example  For an example input see
/course/cs258/data/green/gz_16_1

For an example output see
/course/cs258/data/green/gz_16_1.out

Instructions  We will run your submission using the command: ./gz <timelimit> <filename>

For example: ./gz 60 /course/cs258/data/green/gz_16_1
means the program will use gz_16_1 as input and will run at most 60 seconds.

We use stdout for output. Output to other streams will be ignored. Your submission will be tested on a department linux machine. If your algorithm is a standalone program, please name it gz, otherwise, please specify the compilation procedure and provide a script that follows the above format to run the program.
Resources  Some sample input files are available in: /course/cs258/data/green

4 Remarks

Handin  Command: /course/cs258/bin/cs258_handin hw3
All of the files in the current directory and sub-directories will be submitted. Only the last submission will be marked.

Questions  Please contact the class GTA Carleton (cjc@cs.brown.edu).

Time Limit  You can assume that the time limit will be something between 10 and 600 seconds.

Hint  An optimal value for $g_{100,1}$ is $\approx 19240822$.

Warning  This is a first time assignment, be prepared for clarifications and bug-fixes.