CS-2580: Hw6 Operation Hurricane

Prototype Due Date: Sunday, April 17, 11:59pm

Final Due Date: Sunday, April 25, 11:59pm

1 Problem Statement

A huge hurricane recently decimated Florida and you are in charge of coordinating the infrastructure recovery efforts. At your disposal you have, \( w \), different work crews. Each crew is specialized at repairing one specific type of infrastructure (e.g. waste water, natural gas, electric power, debris clearing, telecommunications, transportation, ...). Intelligence gathers have identified, \( s \), sites that require extensive infrastructure restoration and the repairs at each site require significant coordination. For example, nothing can be repaired until the debris is cleared, then all the infrastructure below the ground should be repaired before above ground infrastructures. Finally, some site specific repairs can be accomplished by more than one of the specialized crews. For example, telephone lines can be repaired by either a communications specialist or an electric power specialist. All of these requirements are represented in the following way. Each site \( i \in 1..s \) has a list of \( t_s \) tasks. Each task \( j \in 1..t_s \) represents the restoration of one infrastructure and has two properties, \( R_{ij} \) the set of crews that are capable of completing task \( j \) at site \( i \) and \( D_{ij} \) the amount of time that is required to complete task \( j \) at site \( i \). Additionally, the tasks must be completed in order. That means task \( j \) must be completed before task \( j + 1 \). A solution to this infrastructure recovery problem has two parts, first \( S_{ij} \), is the start time of every task \( j \) at site \( i \) and \( C_{ij} \) the crew assigned to that task task \( j \) at site \( i \). Keep in mind, that each crew can only work on 1 task at a time and the tasks must be completed in the specified order. The recovery operations remain at full capacity until all the tasks are completed, so the objective in the problem is to minimize the time of the last job’s completion, \( \min S_{ij} + D_{ij} \).

2 Assignment

Write a algorithm to solve the Operation Hurricane Problem. You can apply any technique you want, 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. 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 compile and 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 input file contains $\sum_i t_i + s + 1$ lines, where $s$ is the number of sites to be restored and $t_i$ is the number of tasks at site $i$.

The first line contains two integers $w$ and $s$. It is followed by $s$ blocks, each block starts with the number of tasks at site $i$, $t_i$. It is followed by $t_i$ task lines. Each task line contains, first the number of crews capable of completing the task $|R_{ij}|$ followed by that many crew identifier values$^1$. The line ends with the amount of time all the crews must work on this task, $D_{ij}$. The number of values in each task line is $|R_{ij}| + 2$. All the values are integers.

[Input Format]
w s
t_0
r_0_0 c_0_0_0_1 c_0_0_0_2 ... c_0_0_x d_0_0
r_0_1 c_0_1_0_1 c_0_1_0_2 ... c_0_1_x d_0_1
...
t_s-1
r_s-1_0 c_s-1_0_0_1 c_s-1_0_0_2 ... c_s-1_0_x d_s-1_0
r_s-1_1 c_s-1_1_0_1 c_s-1_1_0_2 ... c_s-1_1_x d_s-1_1

The output file contains $2s + 1$ lines. The first line has two integers $lt$ and $opt\_flag$. $lt$ is the time of the last task completion in the solution. $opt\_flag = 1$ if the algorithm can prove the optimality and 0 otherwise. The $s$ following lines define the the start times of each task at each site. After the start times, the remaining $s$ lines define the crews used for each task at each site.

[Output Format]
lt opt\_flag
s_0_0 s_0_1 s_0_2 ... s_0_t-1
s_1_0 s_1_1 s_1_2 ... s_1_t-1
...
s_s-1_0 s_s-1_1 s_s-1_2 ... s_s-1_t-1
c_0_0 c_0_1 c_0_2 ... c_0_t-1
c_1_0 c_1_1 c_1_2 ... c_1_t-1
...
c_s-1_0 c_s-1_1 c_s-1_2 ... c_s-1_t-1

Example For an example input see
/course/cs258/data/hurricane/hr_001

For an example output see
/course/cs258/data/hurricane/hr_001.out

$^1$Crew identifiers are 1-based, not 0-based.
**Instructions**  We will run your submission using the command: `.hr <timelimit> <filename>

For example: `.hr 300 /course/cs258/data/hurricane/hr_001
means the program will use hr_001 as input and will run at most 300 seconds.

We use stdout for output. Output to other stream will be ignored (you may want to send runtime information to stderr). Your submission will be tested on a department linux machine. If your algorithm is a standalone program, please name it hr, otherwise, please specify the compilation procedure, it is appreciated if you also provide a script that follows the above format to run the program.

**Resources**  You can find operation hurricane instances in /course/cs258/data/hurricane and an example output file, /course/cs258/data/hurricane/hr_001.out

4 Remarks

**Handin**  Command: /course/cs258/bin/cs258_handin hw6
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).

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