Logical reasoning and programming, task I

(October 14, 2018)

Problem

Your task is to solve a slightly modified[1] version of the open job scheduling problem using a SAT solver.

We have t units of time (makespan), m machines, n standard jobs plus a machine maintenance job, and an $(m \times (n+1))$ matrix of non-negative integer weights $W = (w_{ij})$ where w_{ij} is the amount of uninterrupted time units job j requires on machine i (possibly zero) and job $n+1$ is the machine maintenance job. The problem is to decide whether you can schedule all the jobs on machines according to W in such a way that they are completed in t units of time. Moreover, the following two conditions must be satisfied

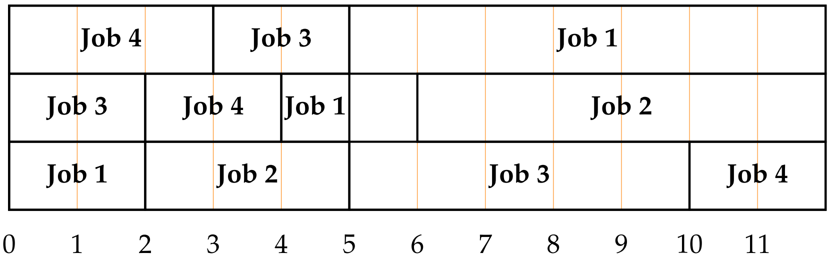
- each machine can process only one job at a time and
- each job may be processed only by one machine at a time, the only exception is the machine maintenance job that can be processed simultaneously on more machines.

Example

For $t = 12$, $m = 3$, $n = 3$, and

$$W = \begin{bmatrix} 7 & 0 & 2 & 3 \\ 1 & 6 & 2 & 2 \\ 2 & 3 & 5 & 2 \end{bmatrix}$$

is the answer yes, because there exists a schedule, e.g.,



where Job 4 is the maintenance job. There exists no schedule for $t = 11$.

Program

You should upload an archive to BRUTE that contains an executable script `openshop` that expects an input string on `stdin` and produces a solution to `stdout`.

It is expected that you use Python (use `python2` or `python3`), but MATLAB 9.2 (use `matlab`) should also work. You may use

[1] Our version contains a machine maintenance.

- PycoSAT in Python, `import pycosat`,
- MiniSat, command `minisat`,
- PicoSAT, command `picosat`,

as SAT solvers. You are allowed to use another solver included in your archive.

Every input has a maximum execution time attached, however, the given time should be enough for solving the problem using a SAT solver with a decent (non-optimized) encoding.

Non-standard settings can be discussed individually.

Input

It is a string containing a sequence of non-negative integers separated by commas. In our example, it is

```
12, 3, 3, 7, 0, 2, 3, 1, 6, 2, 2, 2, 3, 5, 2
```

where the meaning is t, m, n, w_{11}, w_{12}, $\ldots$, $w_{m(n+1)}$. Hence the sequence contains $3 + (m * (n+1))$ numbers.

Output

It is a string of $m * (n+1)$ non-negative integers separated by commas s_{11}, s_{12}, $\ldots$, $s_{m(n+1)}$, where s_{ij} says that job j starts on machine i at time $0 \leq s_{ij} < t$. If $w_{ij} = 0$, then $s_{ij} = 0$ or any other value.

In our example, you are supposed to produce, e.g.,

```
5, 0, 3, 0, 4, 6, 0, 2, 0, 2, 5, 10
```

If no solution is possible, then just produce string

```
0
```