VRt Contest 2019 (marathon)

A2. Collaboration

time limit per test: 15 seconds
memory limit per test: 768 megabytes
input: standard input
output: standard output

Consider several locations in integer points on the plane. One of the locations is the base where workers wait. Every other location contains one job.

Each job has the following characteristics: work duration, the required number of workers, and also the earliest possible moment of starting the job and the latest possible moment of completing it. In order to complete the job, the required number of workers must simultaneously start working on it, and then be continuously present at the location for the duration of the job. Each job must either be fully done exactly once, or left untouched.

Our task is to determine what should the workers do so that we have more profit. any worker can contribute to any job. We can use as many workers as we like, but each worker must take part in doing at least one job.

Each worker starts at the base at a certain moment, then moves between locations and does the jobs, and finally gets back to the base and stops completely. A worker can move along coordinate axes, and spends one minute increasing or decreasing any one of his coordinates by 1. We have to pay the worker 240 credits for appearing in our solution, and additionally 1 credit per minute, up to the moment when the worker stops completely.

We get rewarded for each job that is done. For a job with duration d and the required number of workers p, we get a reward of $d \cdot p \cdot (p + 5)$ credits. The jobs which are not done don't contribute to this sum.

Input

The first line contains an integer n from 500 to 2000: the number of locations. Each of the following n lines contains six integers and describes one location in the format $x\ \ y\ \ d\ \ p\ \ l\ \ h$. The coordinates x and y are integers from 0 to 100. The duration d is an integer from 5 to 30 minutes. The required number of workers p is an integer from 1 to 7 people. The earliest possible start moment l and the latest possible completion moment h are integers from 200 to 800 minutes, chosen so that $60 \le h - l \le 300$.

The first location is the base. It is an exclusion to the rules above and has the format $x\ \ y\ \ 0\ \ 0\ \ 0\ \ 0$. All given points (x, y) are distinct.

In all tests, the number n, as well as all points, durations, numbers of required workers, and time spans are chosen randomly, independently, and uniformly from the available choices.

Output

Print instructions for each worker as a block of a few lines which look as follows:

- `start (at-moment) (at-location)`
  The first line of a block. Determines the moment when the worker starts at the base. The location must be the base, and so have the number 1.
- `arrive (at-moment) (at-location)`
  We want the worker to arrive at this moment to this location. If he can not make it in time after the previous command, the solution is considered incorrect. Locations are numbered from 1 to n in input order.
- `work (start-moment) (end-moment) (at-location)`
  We want the worker to do the job at this location in this exact time interval. The location must be equal to the location in the previous arrival command. If the worker can not make it in time after the previous command, or the given time interval has wrong length or exceeds the time span of the location, the solution is considered incorrect. This command must be present at least once in each block. Recall that each job must be either completed simultaneously by the required number of workers, or left untouched, otherwise, the solution will be considered incorrect.
- `end`
  The last line of a block. The worker must be in the same position where he started.

At each moment, there may be arbitrarily many workers in each location. All moments in the output must be integers from 0 to 1000, and all location numbers must be integers from 1 to n.

Print consecutive blocks one under the other, with no gaps.

Example

Input

```
4
5 15 0 0 0 0
2 13 30 2 200 400
3 12 29 1 350 600
39 21 9 4 671 757
```

Output

```
start 335 1
arrive 340 2
work 340 370 2
arrive 372 3
work 372 401 3
arrive 406 1
end
start 335 1
arrive 340 2
work 340 370 2
arrive 375 1
end
```

Explanation

The reward for the job at location 2 is $30 \cdot 2 \cdot (2 + 5) = 420$ credits.

The reward for the job at location 3 is $29 \cdot 1 \cdot (1 + 5) = 174$ credits.

The job at location 4 is not completed.

We have to pay $240 + 71 = 311$ credits to the first worker.

We also have to pay $240 + 40 = 280$ credits to the second worker.

Profit: $420 + 174 - 311 - 280 = 3$.

Note that this example is too short ($n < 500$) and so does not appear in the testing system. It is included for explanatory purposes only.

Scoring

Each test is scored independently. The score for a test is the profit divided by 1000. It the profit is negative, the score is equal to zero. If a solution did not output a valid answer on a certain test, the result for this test is also zero.

Testing

Your solution will be checked on sets of tests generated in advance. A solution's score is the sum of its results on all the tests. If a solution did not output a valid answer on a certain test, the result for this test is zero.

During the main phase of the contest, there are two ways to submit a solution.

- The first one, problem A1, is to check on examples. There are 10 example tests which are also available for local testing. As soon as the solution is checked, you can see reports for all examples by clicking on the submission result. The tests for A1 are open, you can download them from the links https://assets.codeforces.com/rounds/1160/codeforces-vrt-2019.tar.gz (for Linux) and https://assets.codeforces.com/rounds/1160/codeforces-vrt-2019.zip (for Windows). After testing, you can click the verdict of your submission to see the result for each test.
- The second way, problem A2, is to check on preliminary tests. There are 100 preliminary tests which are generated in advance but kept secret. The score for preliminary tests (but not for example tests) is used in the preliminary scoreboard. This score does not affect the final results, but nevertheless allows to roughly compare a solution with others.

After the main phase ends, for each participant, the system chooses the final solution:

- consider all solutions sent for **preliminary testing**;
- choose the ones which got a total score strictly greater than zero;
- define the final solution as the one of chosen solutions which has **the latest** submission time.

Note that the solutions sent only to be checked on examples are not considered when choosing the final solution.

During final testing, all final solutions will be checked on the same set of a large number (≈ 1000) of final tests. The score for final tests determines the final scoreboard. The winner is the contestant whose solution gets the highest total score. In case two or more participants have equal total score, the contestants with such score tie for the same place.