Problem description

The problem is based on a problem faced by a fictional international company responsible for delivering and installing vending machines. The machines must be delivered within a customer-dependent delivery window and must be installed by a technician as soon as possible after delivery.

The planning horizon consists of a period of consecutive days, numbered 1, 2, and so on. There are different kinds of machines, each having its own size, expressed in the same unit.

There are machine requests from customers that all have to be satisfied. A request consists of a number of machines of one kind, and a delivery window within which these machines have to be delivered. Each delivery window consists of a number of consecutive days within the planning horizon. If a customer needs more than one kind of machine, a separate request is defined for each kind. For example, there can be a request for two machines of kind 5 that have to be delivered on day 8, 9 or 10.

Machine delivery. There is one depot location where all the machines are located at the start of the planning horizon. There are enough machines of each kind available to satisfy all the requests. Trucks are hired to transport the machines from the depot to the customers. There is no limit on the number of trucks that can be hired. All trucks are identical. A truck can accommodate any combination of machines whose total size does not exceed the truck's capacity, where the capacity is expressed in the same unit as the machine sizes. The delivery of a request cannot be split, i.e., all machines of a request must be delivered simultaneously by one truck. In the provided instances, none of the request sizes exceeds the vehicle capacity.

The daily route of a truck starts and ends at the depot, i.e., a truck that picks up a machine from the depot on a certain day must end its route at the depot that same day. A truck can return to the depot several times during a day to pick up machines. The total distance a truck can travel per day is limited to a given maximum. Loading a machine at the depot and unloading a machine at a customer does not take any time.

Machine installation. After delivery, each machine must be installed by a technician at the customer location. Installation of a machine cannot take place on the day the machine is delivered. For every full day a machine is 'idle', i.e., delivered at the customer but not yet installed, a fixed penalty is charged. This penalty is specified for each kind of machine. For example, suppose a machine of type A is delivered on day 4 and installed on day 7. Then the penalty for machine type A is charged twice, as the machine was idle on days 5 and 6. If a machine is delivered on day 4 and installed on day 5, no penalty is charged.

Each technician has a skill set that determines which kinds of machines he/she can install. The maximum number of consecutive days a technician can work is 5. If a technician has worked for five consecutive days, he/she must have two days off. If a technician has worked for less than five consecutive days, a single day off suffices. The daily route of a technician must start and end at his/her home location. The total distance a technician can travel per day is limited to a given maximum. Also the number of installations per day is limited for each technician. Installing a machine does not take any time.

In order to determine the traveled distances, integer coordinates are provided for the depot, the customer locations, and the technician's home locations. The distance between coordinates (x1,y1) and (x2,y2) is defined i.e., the ceiling of the Euclidean distance.

$$\left\lceil \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \right\rceil$$

In a solution all requested machines are delivered and installed within the planning horizon. The objective is to minimize the total cost. There are costs per unit of distance travelled by truck, for using a truck for a day, and for using a truck at all during the planning horizon. Additionally, there are costs per unit of distance travelled by a technician, for using a technician for a day, and for using a technician at all during the planning horizon. Finally, there are costs for every full day a machine is idle, specified for each kind of machine.

Format of the instance data

We now describe the format of the instance data, which is available in text format. Within each section, the entries are sorted by their (consecutive) IDs. We point out that all IDs in the input are positive integers, and that the depot has location ID 1 in every instance.

Example instance

An example of an instance file in text format follows below.

```
DATASET = …
NAME = testInstance

DAYS = 50
TRUCK_CAPACITY = 6
TRUCK_MAX_DISTANCE = 25000

TRUCK_DISTANCE_COST = 1
TRUCK_DAY_COST = 100
TRUCK_COST = 100000
TECHNICIAN_DISTANCE_COST = 1
TECHNICIAN_DAY_COST = 100000
TECHNICIAN_COST = 100

MACHINES = 4
1 1 200
2 1 200
3 2 500
4 1 100
LOCATIONS = 9
1 10 50
2 20 10
3 50 5
4 33 7
5 40 40
6 70 40
7 1 35
8 10 5
9 25 60

REQUESTS = 7
1 2 1 10 1 1
2 3 5 25 3 1
3 4 20 30 4 2
4 5 25 45 1 3
5 6 40 45 2 1
6 7 15 30 4 1
7 7 10 15 2 4

TECHNICIANS = 5
1 3 10000 2 0 1 1 1
2 8 10000 1 1 1 0 0
3 8 10000 5 0 0 1 0
4 9 10000 2 1 1 0 1
5 1 500 1 1 1 1 1
```

Explanation of example instance

Here we give an explanation of the input file. The different sections in the text file will always appear in this given order. However, additional line breaks or spaces may be present. Entries in a line are separated by spaces.

`DAYS` (integer) gives the number of days in the planning horizon. Recall that the days are numbered 1, 2, and so on.
`TRUCK_CAPACITY` (integer) denotes the capacity of one truck.
`TRUCK_MAX_DISTANCE` (integer) is the maximum distance that one truck can travel on a day.

The next section defines the weights for the costs related to the delivery trucks and the technicians.

`TRUCK_DISTANCE_COST` (integer) is the cost per unit of distance traveled by truck.
`TRUCK_DAY_COST` (integer) denotes the cost per route, i.e., the cost per used truck per day. `TRUCK_COST` (integer) is the cost for using a truck during any day of the planning horizon. This cost should be multiplied by the maximum number of trucks needed on a day.
`TECHNICIAN_DISTANCE_COST` (integer) is the cost per unit of distance traveled by a technician.
`TECHNICIAN_DAY_COST` (integer) is the cost per used technician per day.
`TECHNICIAN_COST` (integer) is the cost for using a technician during any day of the planning horizon. This cost should be multiplied by the maximum number of technicians used on a day.

In this particular example, minimizing the maximum number of trucks needed during any one day is far more important than minimizing the total number of routes. For example, using four trucks on every day of the planning horizon (at a cost of 4·50·100+4·100.000 = 420.000) is much better than using five trucks on day 1, two trucks on day 2, and four trucks on every other day of the planning horizon (at a cost of (5+2+4·48)·100+5·100.000 = 519.900). For the technicians, it is the other way around in this example: using four technicians on every day of the planning horizon (at a cost of 4 · 50 · 100.000 + 4 · 100 = 20.000.400) is worse than using five technicians on day 1, two technicians on day 2, and four technicians on every other day of the planning horizon (at a cost of (5 + 2 + 4 · 48) · 100.000 + 5 · 100 = 19.900.500).

The following three sections start with a name, followed by the number of entries in that section. For example, `MACHINES = 4` indicates that there are 4 machine kinds, the details of which are described in the following 4 lines.

Under `MACHINES`, the different machine kinds are listed. The first entry is the machine kind ID. The second entry gives the size of one machine of this machine kind (integer). The third entry denotes the penalty that is charged for every full day a machine of this kind is idle.

Under `LOCATIONS`, the coordinates of the depot location, the customer locations, and the technicians' home locations are given. The first entry is the location ID, the second and third entries denote the actual x- and y-coordinates, respectively, of the location (integers). Recall that the depot always has location ID 1. It is possible that a technician has location ID 1, too; in that case, the technician is based at the depot. Customers cannot have location ID 1.

Under `REQUESTS`, one can find all machine requests by the customers. The first entry is the request ID. The second entry denotes the location ID of the customer that created this request. The next two entries specify the first and last day of the delivery window for this request. The last two entries give the machine kind ID and the number of requested machines of this kind.

Under `TECHNICIANS`, the different technicians are listed. The first entry is the technician ID. The second entry is the location ID. Note that a technician's location ID can coincide with the location ID of the depot or a customer, and that different technicians can be based at the same location and thus have the same location ID. The third entry denotes the maximum distance the technician can travel per day, the fourth entry denotes the maximum number of installations the technician can execute per day. The remaining entries, one for each machine kind, specify which machines the technician can (1) and cannot (0) install. For example, if there are 4 machine kinds and the last 4 entries of a technician are `1 1 0 1`, then this technician can install machine kinds 1, 2, and 4, but cannot install machine kind 3.

Format of the solution data

The solution file specifies the routes for the trucks and the technicians each day. Solution files should be submitted as text files. We will now explain how it should be constructed.

Example solution

An example of a solution file in text format follows below.

```
DATASET = …
NAME = testInstance

TRUCK_DISTANCE = 13654
NUMBER_OF_TRUCK_DAYS = 5
NUMBER_OF_TRUCKS_USED = 3
TECHNICIAN_DISTANCE = 10388
NUMBER_OF_TECHNICIAN_DAYS = 4
NUMBER_OF_TECHNICIANS_USED = 3
IDLE_MACHINE_COSTS = 4000
TOTAL_COST = 728842

DAY = 1
NUMBER_OF_TRUCKS = 3
1 2 4 0 5 6
 2 93                            1
 NUMBER_OF_TECHNICIANS = 0       4
 DAY =                           4
 NUMBER_OF_TRUCKS =              4

1 7 8
2 3
NUMBER_OF_TECHNICIANS = 2
1 2 5 6

3 4 9

DAY = 3
NUMBER_OF_TRUCKS = 0
NUMBER_OF_TECHNICIANS = 2
1 1 8
2 3 7
```

Explanation of example solution

Here we give an explanation of the data file.

The first two lines `DATASET` and `NAME` refer to the instance that the solution file belongs to. The next section containing `TRUCK_DISTANCE`, `NUMBER_OF_TRUCK_DAYS`, `NUMBER_OF_TRUCKS_USED`, `TECHNICIAN_DISTANCE`, `NUMBER_OF_TECHNICIAN_DAYS`, `NUMBER_OF_TECHNICIANS_USED`, and `COST` is optional. It lists some characteristics of the solution, which make it easy to check the total cost of the solution. These results are also returned by the validator and hence can be checked by the user.

`TRUCK_DISTANCE` (integer) is the total distance traveled by all trucks.
`NUMBER_OF_TRUCK_DAYS` (integer) is the total number of truck days used.
`NUMBER_OF_TRUCKS_USED` (integer) is the maximum number of trucks used on a single day.
`TECHNICIAN_DISTANCE` (integer) is the total distance traveled by all technicians.
`NUMBER_OF_TECHNICIAN_DAYS` (integer) is the total number of technician days used.
`NUMBER_OF_TECHNICIANS_USED` (integer) is the total number of technicians used.
`IDLE_MACHINE_COSTS` (integer) is the sum of the costs resulting from idle machines. In this example, the machine(s) from request `1` are idle for one full day.
`TOTAL_COST` (integer) is the total cost of the solution. This total cost is obtained by multiplying the first six integers given above by their corresponding weights as given in the instance description, summing them, and adding the `IDLE_MACHINE_COSTS`. For example, using the weights as given in the example instance, we find a total cost of $13654 \cdot 1 + 5 \cdot 100 + 3 \cdot 100000 + 10388 \cdot 1 + 4 \cdot 100000 + 3 \cdot 100 + 4000 = 728842$.

After these sections the solution is given, day by day. The section starting with `DAY` first lists by the header `NUMBER_OF_TRUCKS` the number of trucks used this day, followed by the truck routes. Next, the header `NUMBER_OF_TECHNICIANS` indicates the number of technicians used, followed by the technician routes.

The routes for the trucks are explained with the example below:

```
1 2 4 0 5 6
```

- The first integer ‘`1`’ refers to the ID of the truck. Since trucks are identical, we can simply number them by 1, 2, 3,...
- The remaining integers are the sequence of request IDs the truck is going to deliver, where the integer `0` indicates a return to the depot. So in this example, truck 1 first delivers requests `2` and `4`, then returns to the depot to pickup new machines, and finally delivers requests `5` and `6`.

Note as all trucks depart from and must return to the depot at the end of the day, this is not mentioned explicitly in the solution format.

The technician routes are given in a similar way. The first entry refers to the technician ID, the following entries refer to the sequence of the request IDs for the technician’s route. Here too, all technician routes must begin and end at the technician’s home location, which is therefore not mentioned explicitly in the solution format.