Problem: Super Slow Supermarket isn't very Super

The Super Slow Supermarket is trying to cut costs by minimizing the number of employees that are working. In
particular, there is ONLY ONE CLERK checking out customers.
However, to hide the fact that there is really only one checkout line, the store is having customers queue in several
lines. After the cashier finishes helping a customer, he will survey all of the lines that are currently queued. Of all
of the customers at the front of those lines, he'll take the customer who has the fewest number of items. If there
are two customers with the same number of items, he'll take the customer who comes from the smaller line
number. The lines are numbered 1 through 12. It's possible that some of these lines will be empty, in which case
these lines are ignored. The number of seconds the store clerk takes to check out a customer is 30 plus 5 times
the number of items. Thus, if a customer has 8 times, the clerk would check her out in 30 + 8*5 = 70 seconds.

The Problem
You will write a program that reads in information about customers: which line they go to the back of (1 through
12), at what time (in seconds) they enter that line, and the number of items they have, and determines at what
time each customer will check out.

The Input (to be read from an input file assignment2input.txt)
The first line will contain a single positive integer, c (c ≤ 25), representing the number of test cases to process.
The test cases follow.
The first line of each test case will have a single positive integer, n (n ≤ 500,000), the number of customers who
are shopping.
The following n lines will have information about each customer. These n lines will be sorted from earliest event
to latest event. Each of these lines will start with a positive integer, t (t ≤ 109), representing the time, in seconds,
from the beginning of the simulation that the customer steps into a line. This is followed by another positive
integer, m (m ≤ 12), representing which line the customer steps into. This is followed by the name of the customer,
a string of 1 to 9 uppercase letters. The last item on the line will be a positive integer, x (x ≤ 100), representing

the number of items the customer has. It is guaranteed that all of the check in times are unique and that all of the
customer names are unique as well.
The Output (to be written on console. No need to generate an output text file)
For each customer, in the order that they get checked out, print a single line with the following format:
CUSTOMER from line X checks out at time T.
where CUSTOMER is the name of the customer checking out, X is the line they entered to check out, and T is
the number of seconds AFTER the start of the simulation, that they complete checking out. (Thus, this time is the
time they get called to cash out, plus the time it takes them to cash out.)