Computer Science I: Program #5 – Cop and Robbers
(Adapted from 2011 South East Regional Problem I: Moving Points)

Check WebCourses for the due date and time.

Note: THERE IS NO REDO OPPORTUNITY FOR THIS PROGRAM!!!

You are a cop and have the unfortunate circumstance of being the only cop at the scene of a robbery with multiple robbers. Unfortunately, all of the robbers have equally split the loot and are running in different directions. Lucky for you, due to your superior professional training, you are faster than all of the robbers! It's also helpful that the robbers aren't particularly bright. They haven't bothered to bring a getaway car and once they pick a direction to run, they continue running in that direction no matter what. (So, you know exactly where they'll be at any point in time.)

Of course, there are many other crimes to stop, so you would like to catch all of the robbers as quickly as possible. We assume that you have a instantaneous taser, so that once you catch up to a particular robber, you can taze him immediately and he'll remain in that position until another cop comes along and books him. As soon as this happens, you immediately move onto catching the next robber, without losing any time.

The Problem

Assuming that you start at (0, 0), given your speed, as well as the initial positions, speeds and fleeing directions of all of the robbers, determine the minimum amount of time it will you take to catch all of them.

The Input

There will be several robberies to solve. Each robbery begins with two integers, N and C, separated by a space, denoting the number of robbers ($1 \leq N \leq 7$) and the speed of the cop ($0 < C \leq 100$) in units/sec, respectively.

Each of the next N lines will have four space-separated integers, X, Y, D and S. (X, Y) will representing the initial position of that particular robber at time $t = 0$ ($-100 \leq X, Y \leq 100$), D represents the direction of movement in degrees (0 degrees is the positive X axis, 90 degrees is the positive Y axis), and S ($0 \leq S < C$) is the speed of that robber in units/sec. It is assumed that all robbers start moving immediately at time $t = 0$.

All of the input cases will be constructed such that the desired minimum time does not exceed 10000.

The input will end with a line with two 0s.

The Output

For each test case, output a single real number on its own line, representing the least amount of time needed for the cop to catch all of the robbers. Print this number to exactly 2 decimal places, rounded. Any answer within 0.01 of the correct answer will be deemed correct.

Sample Input
```
2 25
19 19 32 10
6 45 133 19
5 10
10 20 45 3
30 10 135 4
100 100 219 5
10 100 301 4
30 30 5 3
0 0
```

Sample Output
```
12.62
12.54
```

Mathematics Background
Given that a cop (or robber) starts at position (x0, y0) at time t0, and that he is moving in the direction D with speed S, then his position at time t0 + T (after T seconds have passed), is:

(x0 + STcos(D), y0 + STsin(D))

If you know where a robber will be at some particular time t0+T and you know where a cop is at time t0, to determine if the cop can catch the robber by time t0+T, calculate the distance, D_{chase} between where the cop is at time t0 and where the robber WILL BE at time t0+T. Then, see if ST, the distance the cop travels in T seconds is greater than or less than D_{chase}. If it is less, it is impossible to catch that robber in that much time. If it is more, than you can catch the robber and might be able to do so faster. When these two values, D_{chase} and ST are very close to one another (say absolute value of the difference less than 0.00000001), then your value for T represents the minimum time it will take to catch that particular robber and that new position represents where the cop will attempt to chase the next robber from.

Implementation Hints
This problem is meant to be an application of the binary search method. The general method of solution is as follows:

1) Try all orderings of catching the robbers. (At most there are 7! of these.)
2) For each ordering, simulate going after the robbers in that fixed order.
3) In simulating, use a binary search over the time to figure out how long it will take you to catch each successive robber. A safe low bound for your search is t=0. A safe high bound for your search is the current distance between you and the robber. Based on the input spec, you are at least 1 unit/sec faster than any robber. So, take this distance and add one to 1, since you are guaranteed to gain at least 1 unit per second on any robber. This will be a safe high bound.
4) Once you get the time to catch one robber, update your position to be where you caught this robber and continue to catch the rest.

Deliverables
Please turn in a single source file, *cop.c*, with your solution to this problem via WebCourses before the due date/time for the assignment. Make sure that your program reads form standard in and outputs to standard out, as previously shown in lab.