# Problem 6—Sphere Reflection

One game that the hyperintelligent pandimensional beings (whose protrusions in this dimension appear to humans to be Frankie Mouse and Benjy Mouse) enjoy playing is Intergalactic Bar Billiards. The idea is to ricochet a particle off a planet so that the particle collides with another particle. (If the planet then pops into a black hole, that's their lookout!) Given the coordinates of the original and target particles, and the size and position of the planet, you are to indicate which point on the planet that the shooter should aim for. The shooter MUST ricochet the particle off the planet; he cannot aim directly for the target particle. If either particle is inside the planet or if the planet is between the particles, then a shot cannot be made, and your program should indicate this as well.

**INPUT SPECIFICATION**, Each data case consists of ten floating point numbers separated by one space. The first three are the *x*, *y*, and *z* coordinates of the sphere. The next is the radius of the sphere. The next three are the *x*, *y*, and *z* coordinates of the original particle. The final three are the *x*, *y*, and *z* coordinates of the target particle. Each case ends in **<EOLN>**. The last data case contains all zeroes and is not to be processed; it merely signifies the end of input.

**OUTPUT SPECIFICATION.** The output cases should be processed in the same order as their corresponding input cases. Each output case is "Case *cs*: Aim for (*x*,*y*,*z*)" where *cs* is the case number and (*x*,*y*,*z*) is the point on the sphere to which the original particle should be aimed. *x*, *y*, and *z* should each be rounded to the nearest tenth, one digit to the right of the decimal point. If there is no point to aim for, the output should read "Case *cs*: Cannot aim". Each output case should be followed by two **<EOLN>** characters.

**SAMPLE INPUT**.

```
0·0·0·4·10·10·10·10·10·-10<EOLN>
5·0·0·5·-10·-10·-10·-10·10·10<EOLN>
0·0·0·1·10·0·0·-10·0·0<EOLN>
0·0·0·0·0·0·0·0·0·0<EOLN>
<EOF>
```

**SAMPLE OUTPUT**.

```
Case·1:·Aim·for·(2.8,2.8,0.0)<EOLN>
<EOLN>
Case·2:·Aim·for·(0.0,0.0,0.0)<EOLN>
<EOLN>
Case·3:·Cannot·aim<EOLN>
<EOLN>
<EOF>
```