

NORTHEAST REGIONAL CHAMPIONSHIP ROUND  
OF THE  
2001-02 ACM INTERNATIONAL COLLEGIATE PROGRAMMING CONTEST  
SPONSORED BY IBM  
WESTFIELD STATE COLLEGE, WESTFIELD, MA  
NOVEMBER 3, 2001

**Problem #7: Stay in the Light**

The town of Fooble is illuminated at night by a series of lights. Each light illuminates a circular area of a specified size. Due to recent energy conservation laws, the lights cannot be left on all the time. Therefore all the lights in town are designed to be on for a fixed number of time units and then turned off for the same time period. This cycle repeats forever (i.e. light is on, light is off, light is on, light is off...).

There are two types of lights used in the town. Type A lights that are on and then off, and type B lights that are off and then on. For example, a type A light with a duty cycle of 10 is on for 10 time units, off for 10, on for 10, off for 10 and so on. A type B light, with the same duty cycle, would be off for 10 time units, on for 10, off for 10, and so on.

The Fooble people are a funny lot. They only travel at night and will only travel from one light to another. It always takes a Foobie exactly one time unit to travel from one light to another, regardless of the distance between the lights. Foobies are deathly afraid of the dark so they never walk from one light to another unless the entire area between the lights is illuminated. They are perfectly happy to leave a light that is currently off, provided that some light (or lights) are currently illuminating the entire area they wish to travel. Foobies are willing to wait at a light that has been turned off and is in the dark, in the hope that some light (perhaps the one they are waiting at) will turn on and illuminate the area they wish to travel. Points of tangency for illuminated circles are considered to be lit.

The town has prepared a listing of light information for its residents. The listing includes the number, model, location, duty cycle, and radius of the area illuminated for each light in town. The lights are numbered starting at 0 and their locations are given in terms of nonnegative integer  $x$  and  $y$  values. All distances are in Fooble units. Foobies work exclusively with nonnegative integer values. The first line in an individual example specifies the total number of lights in town, and the lights are always listed in numerical order.

You are to write a program that given a light list, and the number of a light where a Foobie will start from and the number of the light they wish to travel to, will print out a quickest

path (in terms of time units) that the Foobie must travel to reach their destination. Travel directions are vertical, horizontal, and diagonal. The program must output the sequence of moves the Foobie will take and will also indicate when the Foobie must wait before moving on to the next light. Your program must be able to indicate if the Foobie cannot make the desired trip.

The first line of the sample data is the number of examples to be solved. Examples should be separated by a blank line. A title line specifying the example number and using the verbiage given below should also be included. Finally, there should be a blank line after each example title line.

Sample Input:

```
3
2
0 A 0 0 3 1
1 B 0 2 2 1
0 1
3
0 A 0 0 3 1
1 B 0 2 2 1
2 B 0 4 3 1
0 2
2
0 A 2 2 5 2
1 B 2 5 5 2
0 1
```

Output for Example 1:

```
Wait
Wait
Move 0 to 1
```

Output for Example 2:

```
Wait
Wait
Move 0 to 1
Move 1 to 2
```

Output for Example 3:

```
No path exists
```

