# Problem C – Cyclists versus Clouds

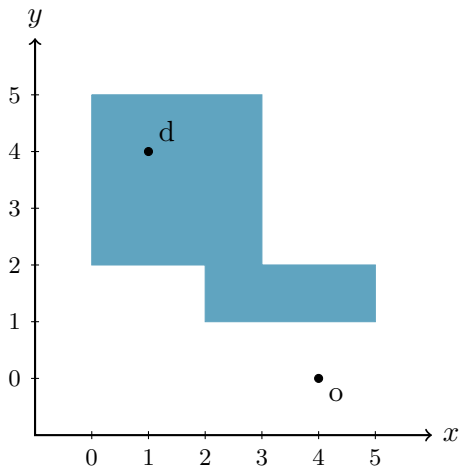
*Author*: Paulo Cezar Pereira Costa, Brasil

In Nlogonia, several campaigns aim to transform the bicycle into the main mean of transportation in the country. One of the actions to promote bikes is a Hackathon focused on the development of applications to facilitate cyclists' day-to-day.

Your university's team has a promising idea. As Nlogonia is a very rainy country, sometimes those who go out in the rain do not intend to get wet but do so purely for lack of option. The idea then is to develop an application capable of generating a route between two points that guarantees that it is possible to make the journey without getting wet, such that there isn't another route with the same guarantee that would take less time.

For the prototype to be presented at the Hackathon, consider that all points of interest are arranged in the Cartesian plane and have integer coordinates. Rain clouds are modeled as simple polygons (non-contiguous edges do not intersect) with each edge parallel to one of the axes. Each cloud moves one unit of distance per unit of time in one of the cardinal directions (North, South, East, or West). A cyclist also moves one unit of distance per unit of time in cardinal directions. The cyclist can change direction at any of the points of interest; changing directions happens instantly. In addition, during the journey the cyclist can stand still at any of the points of interest, protecting themselves from the rain for any integer amount of units of time desired. The cyclist gets wet if they are not stopped at a point of interest (protecting themselves from the rain) and there's at least one cloud above them. A cloud is not considered to be above the cyclist when the cyclist is at its border.

While other team members are concerned with generating the data and creating the graphical interface for the Hackathon, your task is to develop the part of the software responsible for generating the best route for a cyclist who wants to travel between two points of interest without getting wet.




The picture above corresponds to the first two samples. In the first sample, as the cloud moves east, the cyclist can go directly to the destination, first moving west and then north, for a total of seven units of time. In the second sample, the cloud moves south, so the cyclist can only move one unit of distance west without getting wet. If the cyclist tried to move west a second time, the cloud would be above them. The fastest way then is to wait one unit of time after moving west, then move north following the border of the cloud, and finally west towards the destination, for a total of eight units of time.

## Input

The first line contains four integers $X_o$, $Y_o$, $X_d$, and $Y_d$ ($0 \leq X_o, Y_o, X_d, Y_d \leq 100$), indicating that the cyclist starts at point $(X_o, Y_o)$ and wants to arrive to point $(X_d, Y_d)$. The second line

contains an integer $N$ ($0 \leq N \leq 100$) representing the number of rain clouds. After these lines, there are $N$ groups of lines, each group describing a cloud.

Within each group describing a cloud, the first line contains a character $C$ and an integer $V$ ($4 \leq V \leq 100$). The character $C$ is one of the uppercase letters "`N`", "`S`", "`E`" or "`W`", indicating respectively that the cloud moves in the direction North ($y$ ascending), South ($y$ descending), East ($x$ ascending) or West ($x$ descending). The value $V$ represents the number of vertices of the polygon that models the cloud. Each of the next $V$ lines contains two integers $X$ and $Y$ ($0 \leq X, Y \leq 100$), denoting that the point $(X, Y)$ is a vertex of the polygon. Vertices appear in clockwise order. All the given vertices are actual corners of the polygon.

## Output

Output a single line with an integer indicating the minimum amount of units of time needed to travel from $(X_o, Y_o)$ to $(X_d, Y_d)$ without getting wet.

| **Sample input 1** | **Sample output 1** |
|---|---|
| 4 0 1 4<br>1<br>E 8<br>0 5<br>3 5<br>3 2<br>5 2<br>5 1<br>2 1<br>2 2<br>0 2 | 7 |

| **Sample input 2** | **Sample output 2** |
|---|---|
| 4 0 1 4<br>1<br>S 8<br>0 5<br>3 5<br>3 2<br>5 2<br>5 1<br>2 1<br>2 2<br>0 2 | 8 |

| **Sample input 3** | **Sample output 3** |
|---|---|
| 1 2 1 3<br>1<br>N 4<br>1 4<br>2 4<br>2 1<br>1 1 | 1 |

| **Sample input 4** | **Sample output 4** |
|---|---|
| 0 0 0 1<br>1<br>W 4<br>1 1<br>1 0<br>0 0<br>0 1 | 2 |

| **Sample input 5** | **Sample output 5** |
|---|---|
| 20 1 1 10<br>2<br>E 4<br>1 30<br>15 30<br>15 0<br>1 0<br>S 4<br>0 29<br>100 29<br>100 22<br>0 22 | 32 |

| **Sample input 6** | **Sample output 6** |
|---|---|
| 42 42 42 42<br>0 | 0 |