Pokémon Go Go

Purpose: Branch and Bound

Due: December 11th

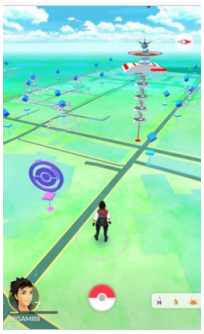

Figure 1: A pokemon go map

Always Catch your Mon, Inc., (also know as ACM), wants to create a new product, called Pokémon Go Go. Users can purchase this application to help them play Pokémon go. The software accesses the pokéstop locations near the current location as well as a list of Pokémon that can be found at each stop. The application then computes the shortest route one can follow to catch all the unique Pokémon, and return to the starting point.

The program assumes that the user is in a city where travel is restricted to moving only in the north–south and east–west directions. The program also assumes that all poké stops are on the intersection of two roads.

For example, consider a case where the application finds five nearby poké stops. Each stop's location is indicated by two integers, (r,c), where r is the number of blocks north of your starting position and c is the number of blocks west of your starting position. Consider if the locations of the five poké stops are $(5, 9)$, $(20, 20)$, $(1, 1)$, $(1, 8)$ and $(2, 8)$ while the names of the Pokémon found at these stops are Evevee, Flareon, Flareon, Jolteon, and Umbreon, respectively. It is clear that one does not have to visit both the second and third stops, since the same Pokémon can be caught at either of them. The best route is to visit the first, fifth, fourth, and third stops (in that order) for a total distance of 28 blocks, since:

- The distance from $(0, 0)$ to $(5, 9)$ is 14.
- The distance from $(5, 9)$ to $(2, 8)$ is 4.
- The distance from $(2, 8)$ to $(1, 8)$ is 1.

- The distance from $(1, 8)$ to $(1, 1)$ is 7.
- The distance from $(1, 1)$ to $(0, 0)$ is 2.

Input

The input holds a single test case. The test case begins with a single integer n, $0 < n \leq 20$, which indicates the number of poké stops to consider. Each of the next n lines specifies the location of a poké stop and the name of a Pokémon that can be found there. The location is specified by two integers r and c separated by a single space, $-10 \leq r, c \leq 10$. The integers r and c indicate that the stop is r blocks north and c blocks east of the starting point. The location is followed by a single space and followed by the string p indicating the name of the Pokémon that can be caught there. Names have between 1 and 25 letters (using only a–z and A–Z). The number of unique Pokémon is always less than or equal to 15. Multiple pokémon can reside at a single poké stop and are listed on separate lines.

Output

Give the shortest distance, in blocks, required to catch all the unique Pokémon.

Sample Input

```
5
5 9 Eevee
10 10 Flareon
1 1 Flareon
1 8 Jolteon
2 8 Umbreon
```

Sample Output

```
28
```

Computing the sample distance

distance from (0,0) to (5, 9) is 5 + 9 = 14
distance from (5, 9) to (2,8) is 3 + 1 = 4
distance from (2,8) to (1,8) is 1 + 0 = 1
distance from (1,8) to (1,1) is 0 + 7 = 7
distance from (1,1) to (0,0) is 1 + 1 = 2
total = 14 + 4 + 1 + 7 + 2 = 28

How the program will be graded

Memo

What	pts
Name	1
Time Analysis O() of every function[1] (in terms of the number of stops	6
Space Analysis O() of every function	6
Test Plan[2] with at least 4 original nontrivial tests [3]	10

Source Code Document

What	pts
Name	1
Description[4]	4
Style	8
pre/post conditions	7
Functionality using the STL	57

[1] The main() is a function.

[2] A test plan is a table with 4 columns and 1 row per test. The columns are named Reason for the test, actual input data, expected output data, and actual output. You do NOT have to have a working program to write a test plan. Each reason should be unique.

[3] A non trivial test contains only legal data (data the conforms to the input specification) with graphs contain at least 1 vertex .

[4] The description should be written to some one who knows NOTHING about the program. It should discuss what the program does (in your own words). After reading the description the user should be able to create legal input and predict the output.