```
#include <iostream>
#include <stdlib>
#include <vector>
#include <string>
#include <ctype>
using namespace std;

void nextMove(int n, String s, String direction , int no_of_moves,  int x, int y){
 if(n == no_of_moves){
   printResult(x , y, direction);
   return null;
 }

 no_of_moves += 1;

 if(s[no_of_moves] == 'L' ||s[no_of_moves] == 'l'){
   var String newDirection = turnLeft(direction);
   nextMove(n, s, newDirection, no_of_moves + 1 ,x ,y);
 }

 if(s[no_of_moves] == 'R' ||s[no_of_moves] == 'r'){
   var String newDirection = turnRight(direction, x, y);
  nextMove(n, s, newDirection, no_of_moves + 1 ,x ,y);
 }

 if(s[no_of_moves] === 'W' ||s[no_of_moves] == 'w'){
```

```
    nextMove(n, s, newDirection,  no_of_moves + 1 ,x ,y);
  }

  if (isInt(s[no_of_moves])){
    switch(direction){
      case : 'SOUTH'
        nextMove(n, s, direction , no_of_moves + 1,x , y-1 )
      break;
      case : 'NORTH'
        nextMove(n, s, direction , no_of_moves + 1,x , y+1 )
      break;
      case : 'EAST'
        nextMove(n, s, direction , no_of_moves + 1,x+1 , y )
      break;
      case : 'WEST'
        nextMove(n, s, direction , no_of_moves + 1,x-1 , y)
      break;
    }
  }
}

String turnLeft(String direction){
  switch(direction){
    case : 'SOUTH'
      return 'EAST';
    case : 'WEST'
```

```
      return 'SOUTH';
    case : 'EAST'
      return 'SOUTH';
    case : 'NORTH'
      return 'WEST';
  }
String turnRight(String direction){
  switch(direction){
    case : 'SOUTH'
      return 'WEST';
    case : 'WEST'
      return 'SOUTH';
    case : 'EAST'
      return 'WEST';
    case : 'NORTH'
      return 'WEST';
  }
}

void printResult(int x ,int  y,String direction){
  cout << x <<" " << y<< "" << direction << endl;
}

int main(int argc, String argv)
{
  int n = argc, r, c;
```

```
  String s = agrv;
  vector <string> v;

  if(argc >= 4){
    cout << "invalid input .";
    return EXIT_FAILURE;
  }

  nextMove(n,s,s[3],0);
  return EXIT_SUCCESS;
}
```