```
import java.util.*;
import java.io.*;
import java.math.*;

/**
 * Auto-generated code below aims at helping you parse
 * the standard input according to the problem statement.
 * ---
 * Hint: You can use the debug stream to print initialTX and initialTY, if Thor seems not follow your orders.
 **/
class Player {

    public static void main(String args []) {
        Scanner in = new Scanner(System.in);
        int lightX = in.nextInt(); // the X position of the light of power
        int lightY = in.nextInt(); // the Y position of the light of power
        int initialTx = in.nextInt(); // Thor's starting X position
        int initialTy = in.nextInt(); // Thor's starting Y position

        // game loop
        while (true) {
            int remainingTurns = in.nextInt(); // The remaining amount of turns Thor can move. Do not remove this line.
            System.err.println(initialTx + " " + initialTy );
              if (Math.abs(lightX - initialTx) > Math.abs(lightY - initialTy)) {
                if (lightX > initialTx && initialTx < 40) {
                    System.out.println("E");
                    initialTx++;
                }    else {
                    System.out.println("W");
                    initialTx--;
                }

  }
            else if (Math.abs(lightX - initialTx) < Math.abs(lightY - initialTy)) {
                if (lightY > initialTy && initialTy < 18) {
                    System.out.println("S");
                    initialTy++;
                }   else {
                    System.out.println("N");
                    initialTy--;
                }
```

```
}
            else if (Math.abs(lightX - initialTx) == Math.abs(lightY - initialTy))   {
               if ((lightX - initialTx) >= 0 && (lightY - initialTy) <= 0)  {
                 System.out.println("NE");
                 initialTy--;
                 initialTx++;
               } else if ((lightX - initialTx) <= 0 && (lightY - initialTy) <= 0)   {
                 System.out.println("NW");
                 initialTy--;
                 initialTx--;
               } else if ((lightX - initialTx) >= 0 && (lightY - initialTy) >= 0) {
                 System.out.println("SE");
                 initialTy++;
                 initialTx++;
               } else {
                 System.out.println("SW");
                 initialTy++;
                 initialTx--;
                 initial Ty--;
               }
            }
      }
   }
}
          // Write an action using System.out.println()
          // To debug: System.err.println("Debug messages...");

 // A single line providing the move to be made: N NE E SE S SW W or NW
```