

TIMUS 1033. Labyrinth Solution

本作品采用[知识共享署名-非商业性使用-相同方式共享 3.0 Unported 许可协议](#)进行许可

write by Gestalti Lur

2012-07-15

题目大意

给出一个 $N \times N$ 的网格迷宫，每个网格的长度为 3，迷宫的四周有墙，内部也有一些障碍物，可以从迷宫的左上角和右下角进入，这两处是没有墙的。可以从这两处开始上下左右向没有障碍的地方移动，计算在迷宫移动时可以接触到的墙的面积(墙高为 3)。

算法分析

请预先了解 floodfill 搜索方法。

从可以进入的两处开始 floodfill. 访问到墙的时候增加计数即可，最后把那两处没有墙的地方减去即可。

参考代码

```
{
floodfill
write by gestapolur
2012-07-09
ACCEPTED
}
program timus1033;
var
  n , cnt : longint;
  w : array[ 0..44 , 0..44 ] of integer;

procedure init();
var
  i , j : longint;
  ch    : char;
begin
  readln( n );
  for i := 1 to n do
  begin
    for j := 1 to n do
    begin
      read( ch );
      if ch = '.' then
        w[ i , j ] := 0
      else
        w[ i , j ] := 1;
    end;
    readln;
  end;
end; { init }

procedure fill(px , py : longint );
begin
  w[ px , py ] := 2;

  if ( px + 1 <= n ) and ( w[ px + 1 , py ] = 0 ) then begin
```

```

    fill( px + 1 , py );
end
else if ( w[ px + 1 , py ] = 1 ) or ( ( px + 1 > n ) ) then
    inc( cnt );

if ( px - 1 > 0 ) and ( w[ px - 1 , py ] = 0 ) then begin
    fill( px - 1 , py );
end
else if ( w[ px - 1 , py ] = 1 ) or ( ( px - 1 = 0 ) ) then
    inc( cnt );

if ( py + 1 <= n ) and ( w[ px , py + 1 ] = 0 ) then begin
    fill( px , py + 1 );
end
else if ( w[ px , py + 1 ] = 1 ) or ( ( w[ px , py + 1 ] = 0 ) ) then
    inc( cnt );

if ( py - 1 > 0 ) and ( w[ px , py - 1 ] = 0 ) then begin
    fill( px , py - 1 );
end
else if ( w[ px , py - 1 ] = 1 ) or ( ( py - 1 = 0 ) ) then
    inc( cnt );

end; { fill }

begin
    init;
    fill( 1 , 1 );
    if w[ n , n ] <> 2 then
        fill( n , n );
    writeln( cnt * 9 - 36 );
end.

```