

Submission

ID	DATE	PROBLEM	STATUS	CPU	LANG
	TEST CASES				
8092548	06:08:29	Dungeon master	✔ Accepted	0.01 s	C++
	✔✔				

Compiler output

```
./dungeon.cpp: In function 'void iniciar(int, int, int)':
./dungeon.cpp:30:14: warning: ignoring return value of 'int scanf(const char*, ...)', declared with attribute warn_unused_result [-Wunused-result]
  30 |         scanf("%c", &car);
      |         ~~~~~^~~~~~
./dungeon.cpp:35:26: warning: ignoring return value of 'int scanf(const char*, ...)', declared with attribute warn_unused_result [-Wunused-result]
  35 |         scanf("%c", &car);
      |         ~~~~~^~~~~~
./dungeon.cpp:43:22: warning: ignoring return value of 'int scanf(const char*, ...)', declared with attribute warn_unused_result [-Wunused-result]
  43 |         scanf("%c", &car);
      |         ~~~~~^~~~~~
./dungeon.cpp:45:18: warning: ignoring return value of 'int scanf(const char*, ...)', declared with attribute warn_unused_result [-Wunused-result]
  45 |         scanf("%c", &car);
      |         ~~~~~^~~~~~
./dungeon.cpp:55:14: warning: ignoring return value of 'int scanf(const char*, ...)', declared with attribute warn_unused_result [-Wunused-result]
  55 |         scanf("%d %d %d", &l, &r, &c);
      |         ~~~~~^~~~~~
```

Submission contains 1 file: [download zip archive](#)

[Help](#)

FILENAME	FILESIZE	SHA-1 SUM	
dungeon.cpp	4319 bytes	b25c021808806e2d7da98915df3f5fcdcc91e03c	download

Edit and resubmit this submission.

dungeon.cpp

```
1  #include <bits/stdc++.h>
2
3  #define INF 999999
4
5  using namespace std;
6
7  struct Punto {
8      int l,r,c;
9
10     Punto(int l1, int r1, int c1) : l(l1), r(r1), c(c1){
11
12     }
13
14     Punto() : l(0), r(0), c(0){
15     }
16 };
17
18 int busqueda_anchura(vector<vector<vector<int>>> dungeon, Punto punto, Punto e, int maxl, int maxr, int maxc);
19
20 void iniciar(int l,int r,int c){
21     char car;
22     Punto inicio, fin;
23
24     int time;
25
26     while(l && r && c){
27
28         vector<vector<vector<int>>> dungeon(l, vector<vector<int>>(r, vector<int>(c, -1)));
29
30         scanf("%c", &car);
31
32         Help for(int ll = 0; ll < l; ll++){
33             for(int rr = 0; rr < r; rr++){
34                 for(int cc = 0; cc < c; cc++){
```

```

35         scanf("%c", &car);
36         if(car == '#')
37             dungeon[l1][rr][cc] = -2;
38         if(car == 'S')
39             inicio = {.l = l1, .r = rr, .c = cc};
40         if(car == 'E')
41             fin = {.l = l1, .r = rr, .c = cc};
42     }
43     scanf("%c", &car);
44 }
45     scanf("%c", &car);
46 }
47
48     time = busqueda_anchura(dungeon, inicio, fin, l, r, c);
49
50     if(time < 0)
51         cout<<"Trapped!"<<endl;
52     else
53         cout<<"Escaped in "<<time<<" minute(s)."<<endl;
54
55     scanf("%d %d %d", &l, &r, &c);
56 }
57 }
58
59 bool entrada(int l, int r, int c, int maxl, int maxr, int maxc){
60     return (l >= 0
61         && l < maxl
62         && r >= 0
63         && r < maxr
64         && c >= 0
65         && c < maxc);
66 }
67
68 int busqueda_anchura(vector<vector<vector<int>>> dungeon, Punto punto, Punto e, int maxl, int maxr, int maxc){
69     queue<Punto> cola;
70     cola.push(punto);
71     dungeon[punto.l][punto.r][punto.c] = 0;
72
73     //La cola de puntos tiene que quedar vacía
74     while(!cola.empty()){
75
76         Help Punto punto_aux = cola.front();
77         cola.pop();
78

```

```

79     if(punto_aux.l == e.l && punto_aux.r == e.r && punto_aux.c == e.c)
80         return dungeon[punto_aux.l][punto_aux.r][punto_aux.c];
81
82     if(entrada(punto_aux.l+1, punto_aux.r, punto_aux.c, maxl, maxr, maxc)
83         && dungeon[punto_aux.l+1][punto_aux.r][punto_aux.c] == -1){
84
85         cola.push(Punto(punto_aux.l+1, punto_aux.r, punto_aux.c));
86         dungeon[punto_aux.l+1][punto_aux.r][punto_aux.c] = dungeon[punto_aux.l][punto_aux.r][punto_aux.c]+1;
87
88     }
89
90     if(entrada(punto_aux.l-1, punto_aux.r, punto_aux.c, maxl, maxr, maxc)
91         && dungeon[punto_aux.l-1][punto_aux.r][punto_aux.c] == -1){
92
93         cola.push(Punto(punto_aux.l-1, punto_aux.r, punto_aux.c));
94         dungeon[punto_aux.l-1][punto_aux.r][punto_aux.c] = dungeon[punto_aux.l][punto_aux.r][punto_aux.c]+1;
95
96     }
97
98     if(entrada(punto_aux.l, punto_aux.r+1, punto_aux.c, maxl, maxr, maxc)
99         && dungeon[punto_aux.l][punto_aux.r+1][punto_aux.c] == -1){
100
101         cola.push(Punto(punto_aux.l, punto_aux.r+1, punto_aux.c));
102         dungeon[punto_aux.l][punto_aux.r+1][punto_aux.c] = dungeon[punto_aux.l][punto_aux.r][punto_aux.c]+1;
103
104     }
105
106     if(entrada(punto_aux.l, punto_aux.r-1, punto_aux.c, maxl, maxr, maxc)
107         && dungeon[punto_aux.l][punto_aux.r-1][punto_aux.c] == -1){
108
109         cola.push(Punto(punto_aux.l, punto_aux.r-1, punto_aux.c));
110         dungeon[punto_aux.l][punto_aux.r-1][punto_aux.c] = dungeon[punto_aux.l][punto_aux.r][punto_aux.c]+1;
111
112     }
113
114     if(entrada(punto_aux.l, punto_aux.r, punto_aux.c+1, maxl, maxr, maxc)
115         && dungeon[punto_aux.l][punto_aux.r][punto_aux.c+1] == -1){
116
117         cola.push(Punto(punto_aux.l, punto_aux.r, punto_aux.c+1));
118         dungeon[punto_aux.l][punto_aux.r][punto_aux.c+1] = dungeon[punto_aux.l][punto_aux.r][punto_aux.c]+1;
119
120     }
121
122     if(entrada(punto_aux.l, punto_aux.r, punto_aux.c-1, maxl, maxr, maxc)

```

```
123     && dungeon[punto_aux.l][punto_aux.r][punto_aux.c-1] == -1){
124
125     cola.push(Punto(punto_aux.l, punto_aux.r, punto_aux.c-1));
126     dungeon[punto_aux.l][punto_aux.r][punto_aux.c-1] = dungeon[punto_aux.l][punto_aux.r][punto_aux.c]+1;
127
128     }
129
130     }
131
132     return -1;
133 }
134
135 int main(){
136
137     int l;
138     int r;
139     int c;
140     cin>>l>>r>>c;
141     iniciar(l,r,c);
142
143     return 0;
144 }
```