Alumno:

Calva Hernández José Manuel 2017630201

Análisis de Algoritmos

M. en C. Edgardo Adrián Franco Martínez  
Grupo: 3CM3  
Fecha: 17 / Mayo / 2018

Diseño de soluciones con Programación

VorazEjercicio 08



Instituto Politécnico Nacional

Escuela Superior de Cómputo

Índice

[Scarecrow 2](#_Toc517119470)

[Redacción del ejercicio 2](#_Toc517119471)

[Captura de aceptación 2](#_Toc517119472)

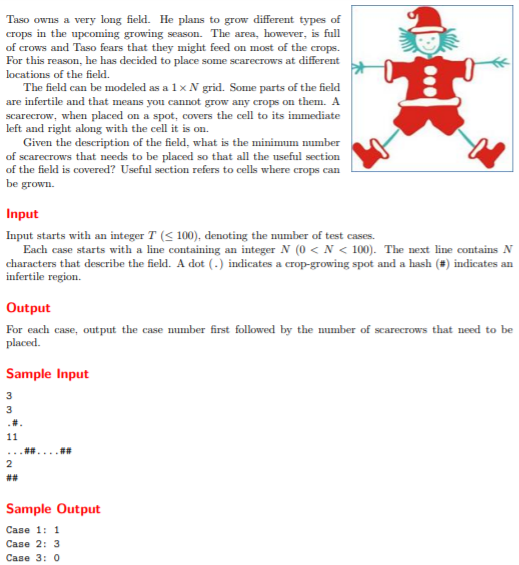
[Explicación de la solución 3](#_Toc517119473)

[Código 3](#_Toc517119474)

[Análisis de complejidad 3](#_Toc517119475)

# Scarecrow

Redacción del ejercicio



Captura de aceptación



Explicación de la solución

Recorreremos cada una de las cadenas siguiendo la siguiente lógica:

* Si ingresan un ‘#’ lo ignoraremos directamente.
* Si ingresan un ‘.’, añadiremos una unidad al resultado, a partir de ello, sabemos que la forma más óptima de colocar ese espantapájaros es cubriendo 3 unidades sin importar qué haya en ellas, por lo tanto, ignoraremos las siguientes dos casillas que nos ingresen prosiguiendo con el algoritmo descrito.

Código

1. #
2. include < bits / stdc++.h > using namespace std;
3. int main(int argc, char
4. const \* argv[]) {
5. int t, n;
6. cin >> t;
7. for (int i = 0; i < t; ++i) {
8. cin >> n;
9. char c;
10. int result = 0;
11. while (n-- > 0) {
12. cin >> c;
13. if (c == '#') {
14. continue;
15. } else if (c == '.') {
16. result++;
17. if (n > 2) {
18. n -= 2;
19. cin >> c;
20. cin >> c;
21. } else {
22. while (n-- > 0) {
23. cin >> c;
24. }
25. }
26. }
27. }
28. cout << result << endl;
29. }
30. return 0;
31. }

Análisis de complejidad

La complejidad es muy sencilla debido a que únicamente tenemos dos datos importantes, n y t, una vez ingresados, haremos iteraciones recorriéndolos de manera lineal en el peor de los casos, por tanto, la complejidad del problema en general viene dada por: