Algoritmo A*

Miembros:

Antonio Cañete Baena

Alejandro Román Sánchez

Antonio Blas Moral Sánchez

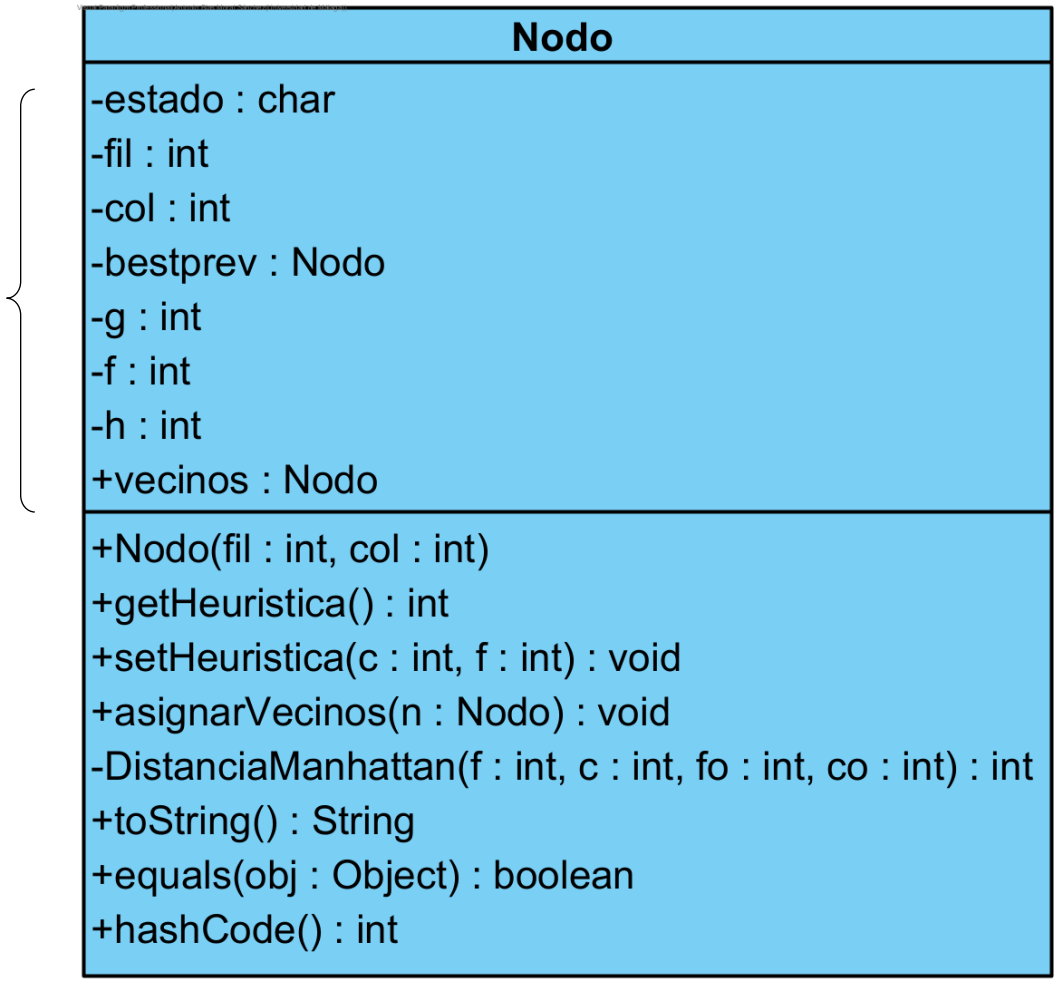
8 de mayo de 2023

Índice

Clase Nodo

En la clase **Nodo** definimos la estructura de datos que vamos a utilizar para cada nodo de nuestro laberinto.

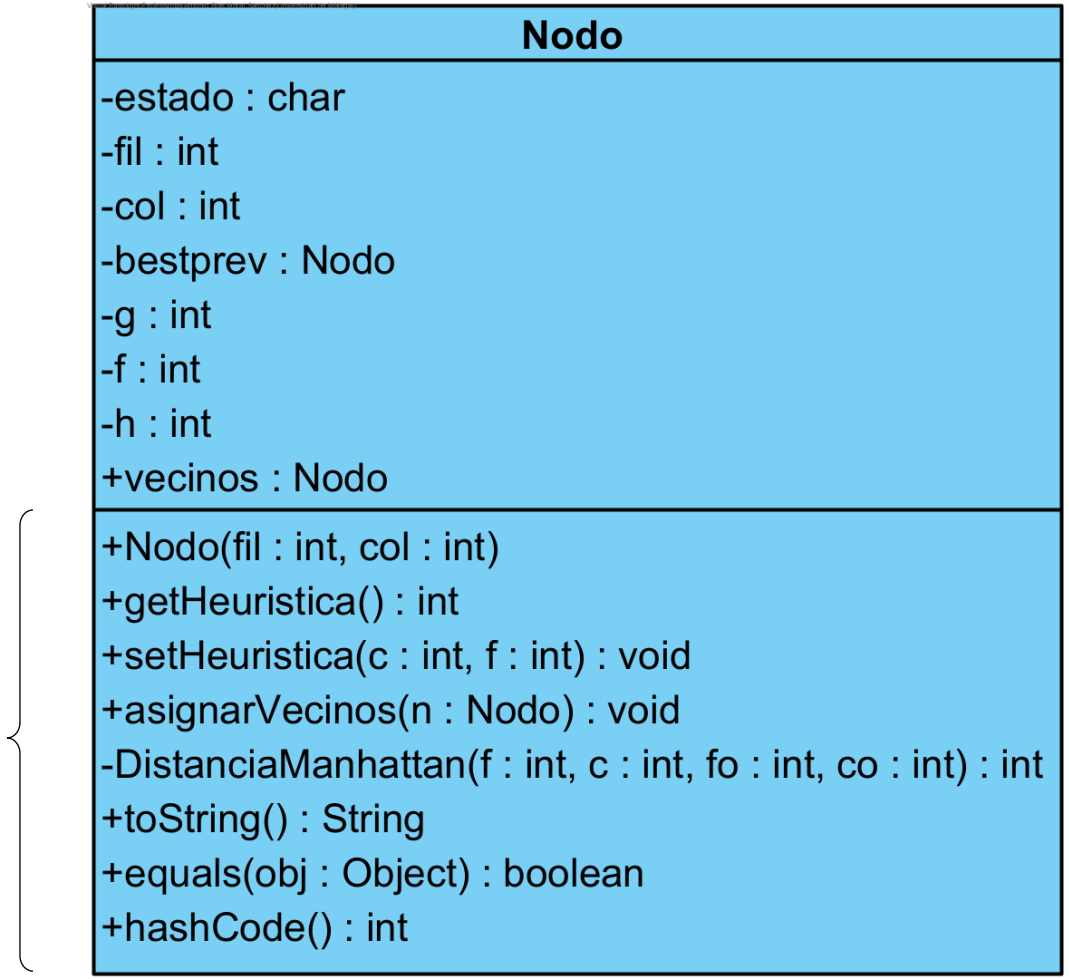
Cada nodo tiene su **estado** (para indicar si es inicial, final o un obstáculo), su fila y su columna dentro del laberinto, los valores de las funciones **f, g** y **h** del algoritmo A*, un listado de los vecinos del nodo, y por último un nodo donde almacenamos el nodo más prometedor para avanzar.



Clase Nodo

Los método implementados dentro de la clase (obviando *getters* y *setters*) son:

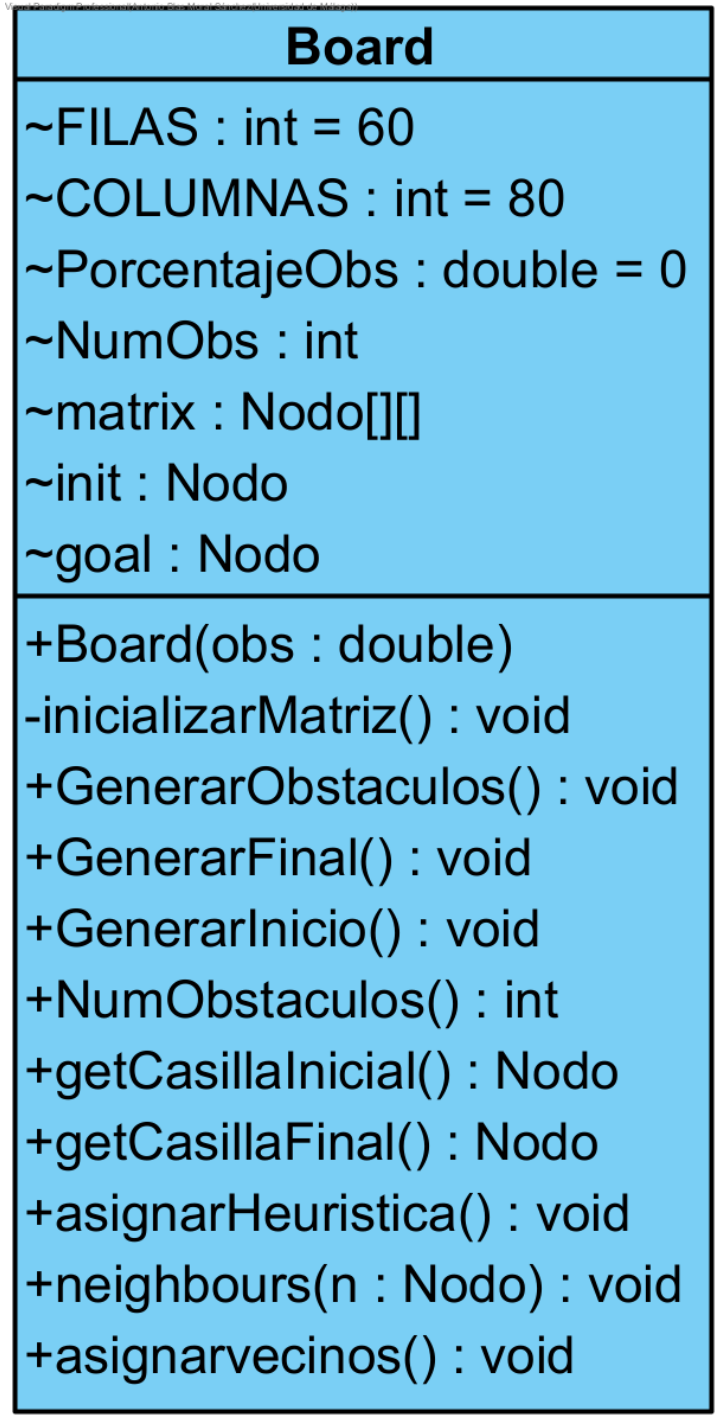
- Distancia Manhattan
- Establecer la heurística del nodo
- Añadir un vecino a la lista de vecinos
- Métodos de clase de java *(equals, toString, ...)*



Clase Board

La clase **Board** representa al laberinto, y esta contiene los siguientes atributos:

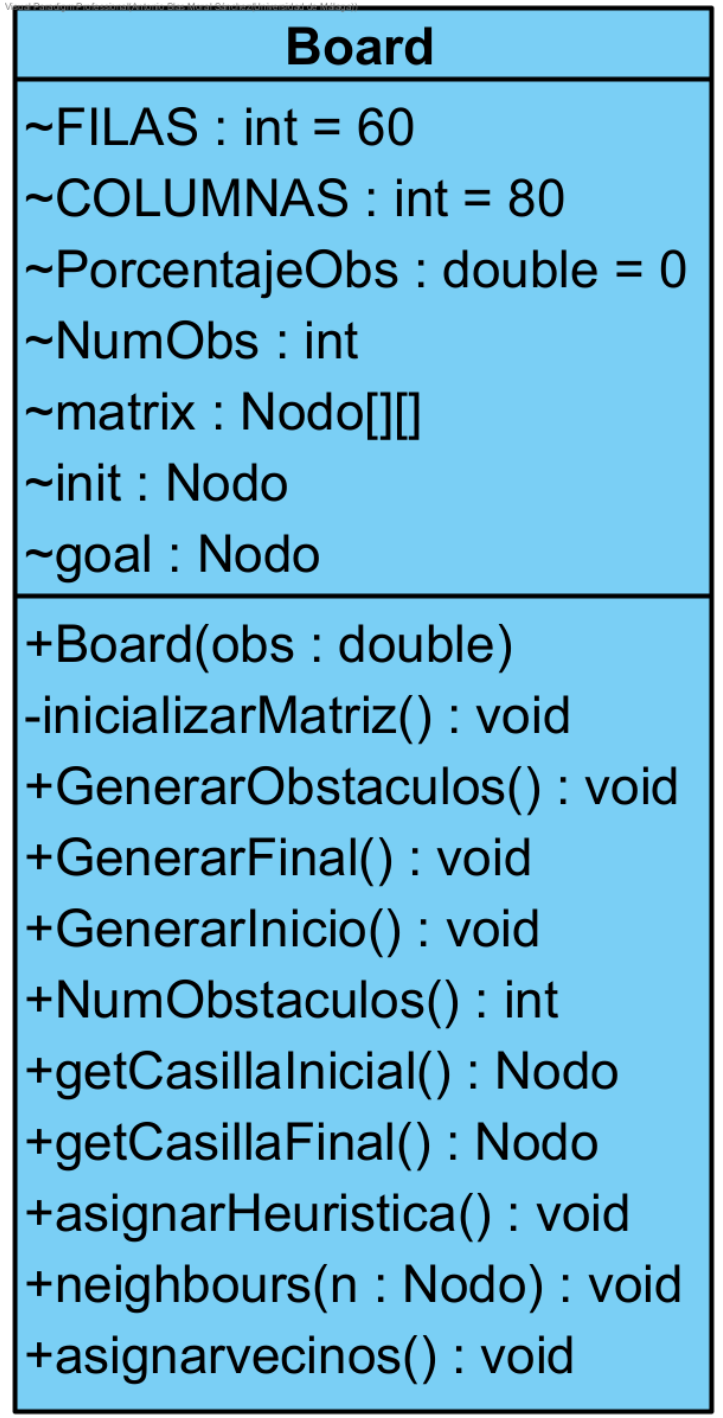
- Tamaño en filas y columnas
- Porcentaje de obstáculos
- Matriz de nodos (representa el tablero)
- Nodo inicial y objetivo



Clase Board

Los métodos implementados dentro de la clase son:

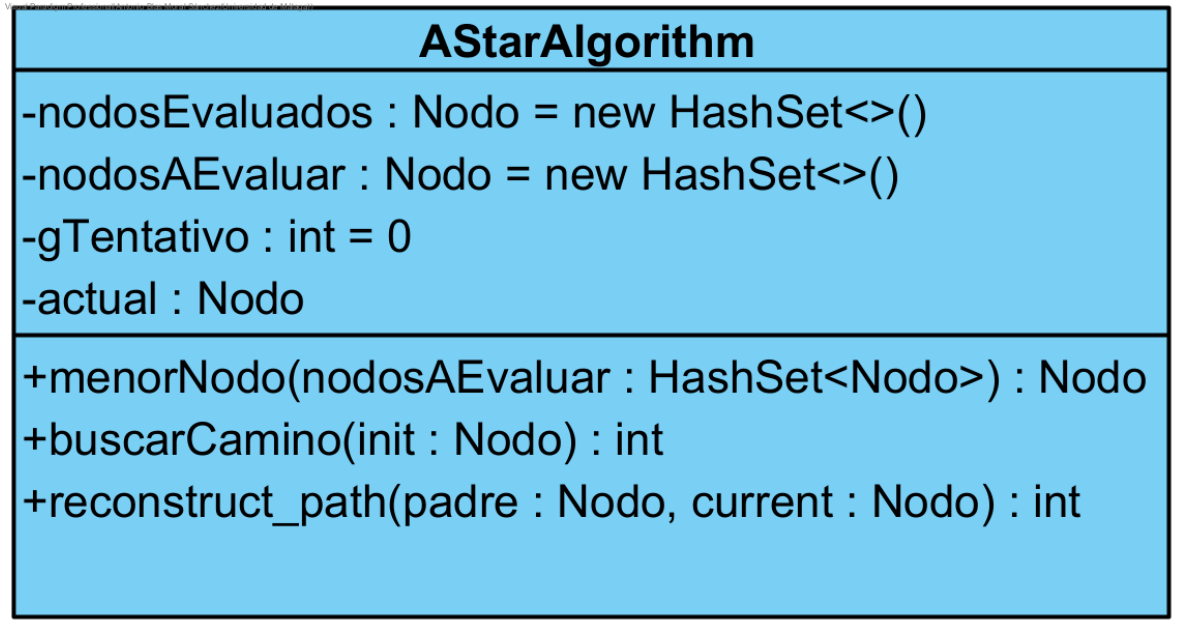
- Inicializar la matriz de nodos.
- Generar los obstáculos(según el porcentaje de obstáculos), la casilla inicial y objetivo de forma aleatoria.
- Asignar la heurística y vecinos a cada nodo del laberinto.



Clase A*Algorithm

La clase **AStarAlgorithm** tiene como atributos dos conjuntos de Nodos (nodos a evaluar y evaluados), el nodo actual y un coste tentativo **g.**

Los método de esta clase son: la búsqueda del nodo con menor valor de **f,** el **algoritmo A*** aplicado a encontrar el camino (pseudocódigo suministrado), la reconstrucción del camino (que modifica el estado del nodo).

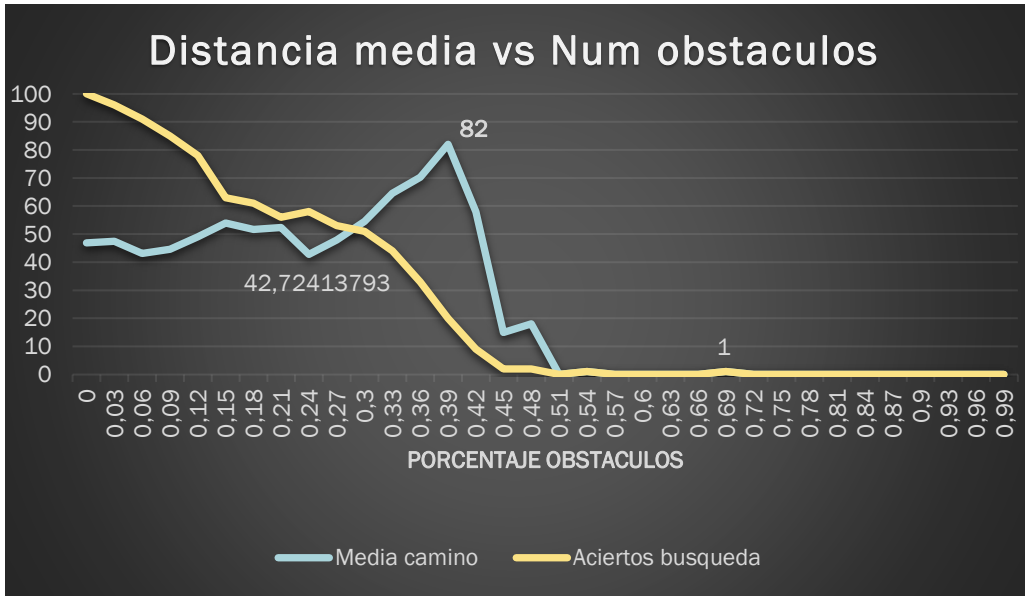


Gráficas

Como podemos observar, cuando nuestro tablero llega a un determinado número de obstáculos, la longitud del camino va disminuyendo considerablemente.

Además, al haber más obstáculos, podemos ver que la probabilidad de encontrar un camino es cada vez mucho menor hasta el punto de que la probabilidad es **nula** puesto que los nodos iniciales o finales no se pueden generar o se generan encerrados entre obstáculos.

Longitud media: 44,41902885

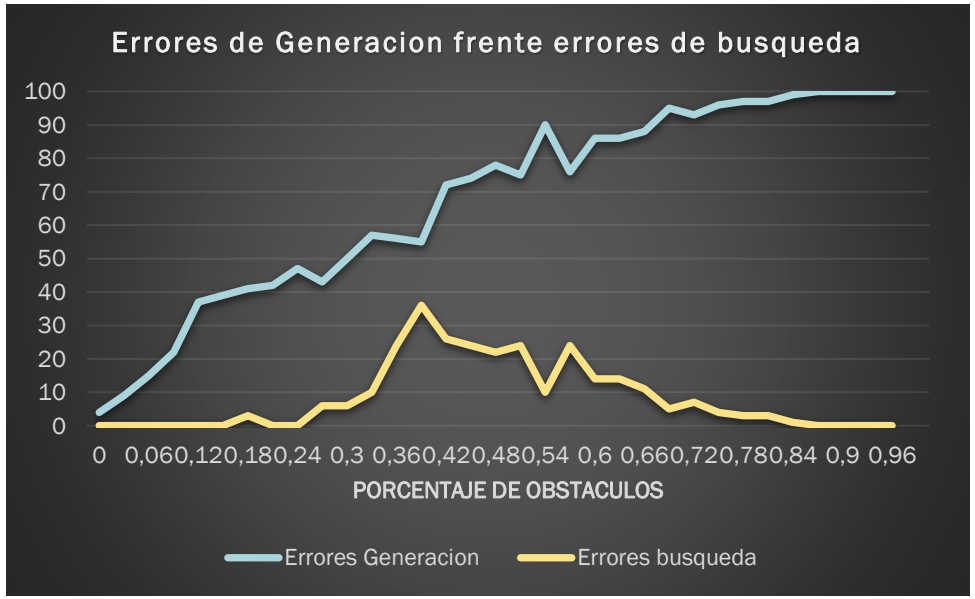


Destacamos algunos cosos con un numero alto de obstáculos en los que se genera un camino valido de longitud muy corta, lo que significa que tanto el nodo inicial como el final se han generado muy cerca

Gráficas

En este caso, en un nivel medio de obstáculos, es cuando mayor es el número de errores por búsqueda debido a que la probabilidad de que el nodo objetivo o el nodo inicial estén encerrados es mayor.

También, podemos observar que a mayor porcentaje de obstáculos, mayor es el número de errores por generación, ya que al haber más obstáculos, es más probable que la casilla inicial o final se coloque en uno de ellos.



Fin

Concluimos la practica aprendiendo un algoritmo muy interesante para muchos ámbitos como los videojuegos, la robótica, sistemas GPS, ...