



UNIVERSIDAD CATÓLICA DEL  
NORTE ESCUELA DE INGENIERÍA,  
SEDE COQUIMBO



**Taller 2**

**Wumpus – Best First**

Nombre: Nicolás Hidalgo Castillo.

Rut:19.302.002-K.

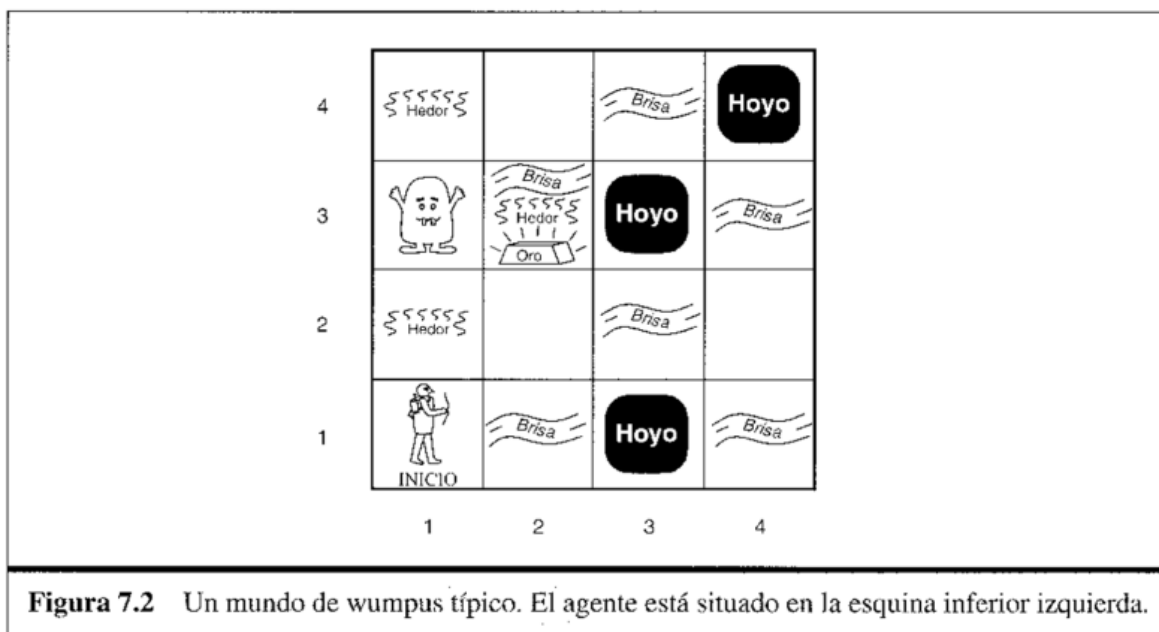
Asignatura: Sistemas inteligentes.

Profesor: Héctor Soza Pollman.

El problema que se desarrolla en el presente taller es el mundo del wumpus propuesto por Rusell y Norvig, esta propuesta consiste en los siguientes puntos:

- En una cueva que está compuesta por habitaciones conectadas mediante pasillos.
- Escondido en algún lugar de la cueva está el wumpus, una bestia que se come a cualquiera que esté en su habitación.
- El wumpus puede ser derribado por la flecha de un agente, y éste sólo dispone de una.
- Algunas habitaciones contienen hoyos sin fondos que atrapan a aquel que deambula por dichas habitaciones (menos el wumpus, que es demasiado grande para caer en ellos).
- Hay habitaciones con una pila de oro.
- Existe habitaciones en que hay un muy malo olor (se debe a un wumpus cerca, este puede estar vivo o muerto).
- Hay habitaciones con una ligera brisa.

En la siguiente imagen se puede observar a modo de ejemplo el cómo se estructura una de las alternativas al problema.



Por último, el algoritmo de búsqueda que se requiere a utilizar es Best First, por lo que se hará uso de una heurística.

**Estado Inicial:** El agente se posiciona en la entrada de la cueva la cual es en la posición (0,0)

**Estado Final:** El agente se posiciona en la habitación donde se encuentra el oro para capturarlo, encontrando la meta.

### **Reglas y restricciones:**

- El agente solo puede moverse vertical y horizontal.
- El agente puede girar a la derecha e izquierda.
- El agente puede percibir 4 estados:
  1. Hedor: El wumpus se encuentra en una habitación adyacente.
  2. Brisa: Existe un hoyo cerca en una habitación adyacente.
  3. Resplandor: El oro se encuentra en la habitación.
  4. Golpe: El agente choca con una pared.
- El agente solo contiene una flecha para disparar.

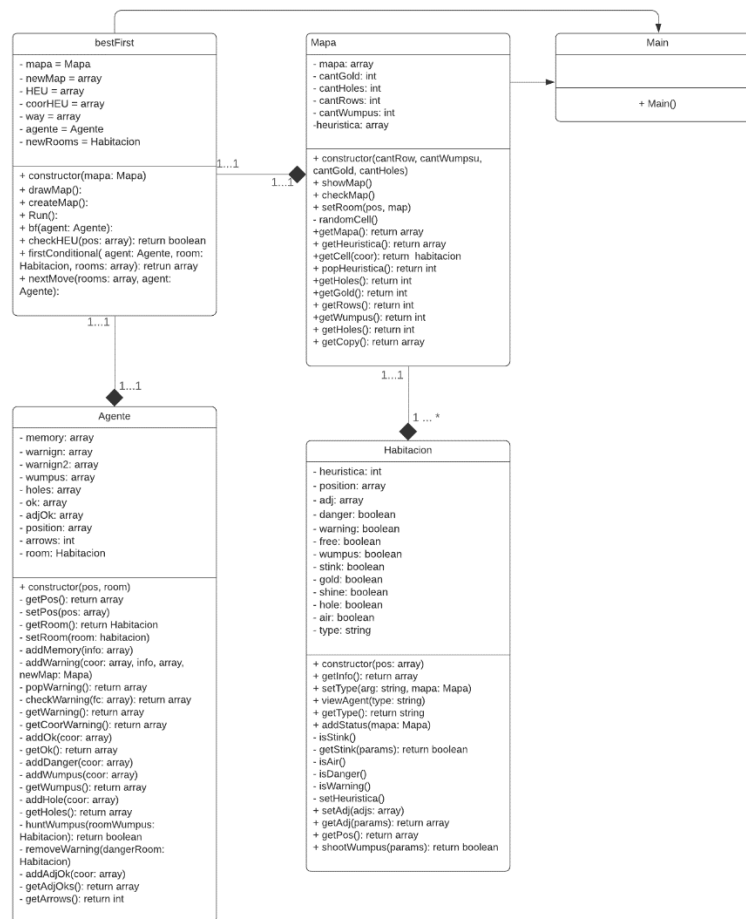
### **Desarrollo del programa.**

El lenguaje que se decidió por utilizar para la creación del programa fue Python con la versión 3.8.1, junto con el uso de visual studio code como IDE, dado esto es que se procedió a desarrollar el código del programa.

Dicho programa se estructuro en base a orientación a objetos, dado esto es que al momento de usar el objeto mapa permite generar una matriz de N dimensiones, dependiendo del parámetro que se le

mande, bajo la misma lógica es que se puede agregar X cantidad de wumpus, Y cantidad de hoyos y Z cantidad deoros esparcidos dentro del mapa, la suma de estas tres ultimas cantidades debe de ser menor o igual a  $N^2$  para sobrepasar las dimensiones agregadas en un comienzo, cabe mencionar que la heurística aplicada se genera aleatoriamente al momento tener el mapa creado en su totalidad.

## Diagrama de clases.



## Clases.

1. Mapa: Representa la cueva en donde se procederá a realizar la búsqueda, este al recibir ciertas variables en los parámetros del constructor, para el correcto funcionamiento de esta depende de habitaciones.
2. Habitación: Simula una habitación de la cueva, esta puede contener un wumpus, un hoyo o el oro que se busca.
3. Agente: Hace representación al agente que se encarga de realizar la búsqueda el oro en la cueva haciendo sus respectivos movimientos permitidos.
4. bestFirst: Esta clase contiene el algoritmo para realizar la búsqueda de best first en donde se hace uso de funciones recursivas para poder proceder con la búsqueda.
5. Main: clase principal que se encarga de ejecutar el programa.

Para las funciones que tienen gran relevancia para el correcto funcionamiento del programa son las siguientes:

1. bf: Esta función se encarga de realizar el recorrido recursivo de la búsqueda que hace el agente.
2. addWarning: Función que se encarga de agregar las coordenadas y tipos de warning que detecte el agente
3. checkWarning: Función que se encarga de detectar el tipo de warning que se detecta en base a las alarmas registradas, también esta función se encarga de detectar aquellas casillas que se encuentran sin riesgo permitiendo eliminarlas de arreglo warning que se tiene.

## **Conclusión**

Como experiencia obtenida en cuanto al algoritmo best first que se llevó a cabo, dado a que para el funcionamiento de este es necesario el uso de una heurística la cual al ser aleatoria, en ciertas ocasiones, la búsqueda del oro es más demorosa debido a que existe la posibilidad de asignar el valor máximo que se tiene en la heurística a la posición donde se encuentra el oro, dado pie así a que el agente recorra todas las habitaciones seguras y dejando como última alternativa aquella que contiene el objetivo a buscar, si bien este caso se puede apreciar, existe la posibilidad de que en dicha habitación se le asigne la heurística menor que se tiene en ese entonces, dando paso así a que el agente pueda encontrar rápidamente el oro. Debido a esto es que el algoritmo es en cierta medida inestable en términos de coste dado a los casos anteriormente mencionados.

Caso 1: El agente no encuentra caminos seguros y se retira.

```
-----START-----
---SIMBOLISMO---
A: AIRE
S: PRESTE
H: HOYO
W: WUMPUS
G: ORO
AG: RECORRIDO DEL AGENTE
SEG: LUGAR SEGURO

['F', 'F', 'F', 'F']
['F', 'H', 'F', 'G']
['F', 'F', 'W', 'F']
['F', 'F', 'F', 'H']

[10, 2, 7, 13, 3, 11, 14, 6, 9, 2, 3, 15, 15, 6, 14, 12]
NEW MAP CREATED
Array: [[2, [0, 1]], [7, [1, 0]]]

['   AG    ', '   SEG    ', '   ', '   ', '   ', '   ']
['   SEG    ', '   ', '   ', '   ', '   ', '   ', '   ']
['   ', '   ', '   ', '   ', '   ', '   ', '   ']
['   ', '   ', '   ', '   ', '   ', '   ', '   ']

EL AGENTE AVANZA HACIA LA CELDA [0, 1]

SE A DETECTADO ['A']
AGREGANDO REGISTROS
EN LA CELDA: [0, 2]
EN LA CELDA: [1, 1]
Array: [[7, [1, 0]]]

['   AG    ', '   AG    ', '[\'A\']', '   ', '   ', '   ']
['   SEG    ', '   \', \'A\']', '   ', '   ', '   ', '   ']
['   ', '   ', '   ', '   ', '   ', '   ', '   ']
['   ', '   ', '   ', '   ', '   ', '   ', '   ']

EL AGENTE AVANZA HACIA LA CELDA [1, 0]

SE A DETECTADO ['A']
AGREGANDO REGISTROS
EN LA CELDA: [1, 1]
EN LA CELDA: [2, 0]
HOYO EN LA CELDA [1, 1]
Array: []

NO SE HAN ENCONTRADO MAS CAMINOS SEGUROS.
['   AG    ', '   AG    ', '[\'A\']', '   ', '   ', '   ']
['   AG    ', '   |H|    ', '   ', '   ', '   ', '   ']
[['\'A\']], '   ', '   ', '   ', '   ', '   ', '   ']
['   ', '   ', '   ', '   ', '   ', '   ', '   ']
```

### Case 2 – El agente detecta donde se encuentra el wumpus y avanza.

```
--START--SIMBOLISMO--A: AIRES: PRESTEH: HOYOW: WUMPUSG: OROAG: RECORRIDO DEL AGENTESEG: LUGAR SEGUROP[ 'F', 'F', 'F', 'F' ][ 'F', 'W', 'F', 'H' ][ 'F', 'F', 'F', 'F' ][ 'H', 'F', 'F', 'G' ]
```

[4, 3, 14, 10, 13, 2, 11, 14, 11, 12, 8, 9, 6, 10, 15, 14]  
NEW MAP CREATED  
Array: [[3, [0, 1]], [14, [1, 0]]]

```
P   AG      , , SEG     , ,          , ,  
P   SEG    , ,           , ,         , ,'  
P       , ,             , ,        , , '  
P       , ,              , ,        , , '
```

EL AGENTE AVANZA HACIA LA CELDA [0, 1]

```
SE A DETECTADO ['S']  
AGREGANDO REGISTROS  
EN LA CELDA: [0, 2]  
EN LA CELDA: [1, 1]
```

Array: [[14, [1, 0]]]

```
P   AG      , , AG      , , ['S'], , '  
P   SEG    , , ['S'], , , , , , '  
P       , ,            , ,      , , '  
P       , ,            , ,      , , '
```

EL AGENTE AVANZA HACIA LA CELDA [1, 0]

```
SE A DETECTADO ['S']  
AGREGANDO REGISTROS  
EN LA CELDA: [1, 1]  
EN LA CELDA: [2, 0]
```

WIMPUS EN LA CELDA [1, 1]

Array: [[10, [1, 1]]]  
  
Array: [[10, [1, 1]]]

```
P   AG      , , AG      , , ['S'], , '  
P   AG      , , |w|      , , , , '  
[[['S'], , , , , , , , '  
P   , ,      , , , , , , , , '
```

EL AGENTE AVANZA HACIA LA CELDA [1, 1]



```

EL AGENTE SE PREPARA PARA DISPARAR A LA CELDA 1 1
RECARGANDO ARCO
EL AGENTE DISPARO HACIA LA CELDA [1, 1]
SE ESCUCHA UN GRITO (EL WUMPUS MUERE)
Array: [[2, [2, 1]], [13, [1, 2]]]

[' AG ', ' AG ', ['S'], ' ' ' ]
[' AG ', ' AG ', SEG ' ' ' ' ]
[['S'], ' SEG ' ' ' ' ' ' ]
[' ' ' ' ' ' ' ' ' ]

```

```

EL AGENTE AVANZA HACIA LA CELDA [2, 1]

SE A DETECTADO ['S']
AGREGANDO REGISTROS
EN LA CELDA: [2, 0]
EN LA CELDA: [2, 2]
EN LA CELDA: [3, 1]

WUMPUS EN LA CELDA [2, 0]

EL AGENTE SE QUEDO SIN FLECHAS
Array: [[13, [1, 2]]]
Array: [[13, [1, 2]]]

[' AG ', ' AG ', ['S'], ' ' ' ]
[' AG ', ' AG ', SEG ' ' ' ' ]
[' |W| ', ' AG ', ['S'], ' ' ' ]
[' ' ' ' ' ' ' ' ' ]

```

```

EL AGENTE AVANZA HACIA LA CELDA [1, 2]

SE A DETECTADO ['S', 'A']
AGREGANDO REGISTROS
EN LA CELDA: [0, 2]
EN LA CELDA: [1, 3]
EN LA CELDA: [2, 2]

WUMPUS EN LA CELDA [0, 2]

EL AGENTE SE QUEDO SIN FLECHAS
Array: []

WUMPUS EN LA CELDA [2, 2]

EL AGENTE SE QUEDO SIN FLECHAS
Array: []

NO SE HAN ENCONTRADO MAS CAMINOS SEGUROS.
[' AG ', ' AG ', |W| ' ' ' ' ]
[' AG ', ' AG ', ' ' ' ' ['S', 'A']]
[' |W| ', ' AG ', ' |W| ' ' ' ' ]
[' ' ' ' ' ' ' ' ' ]

```

### Case 3 – El agente encuentra el oro y se retira.

```

START
---SIMBOLISMO---
A: AIRE
S: PRESTE
H: HOYO
W: WUMPUS
G: ORO
AG: RECORRIDO DEL AGENTE
SEG: LUGAR SEGURO

['F', 'F', 'F', 'F']
['F', 'F', 'H', 'H']
['F', 'F', 'F', 'F']
['G', 'W', 'F', 'F']

[6, 8, 7, 7, 14, 12, 11, 4, 12, 3, 6, 6, 10, 9, 6, 6]
NEW MAP CREATED
Array: [[7, [1, 0]], [8, [0, 1]]]

['   AG    ', '   SEG    ', '   ', '   ', '   ']
['   SEG   ', '   ', '   ', '   ', '   ']
['         ', '         ', '         ', '         ', '   ']
['         ', '         ', '         ', '         ', '   ']

EL AGENTE AVANZA HACIA LA CELDA [1, 0]

Array: [[7, [1, 1]], [8, [0, 1]], [14, [2, 0]]]

['   AG    ', '   SEG    ', '   ', '   ', '   ']
['   AG    ', '   SEG    ', '   ', '   ', '   ']
['   SEG   ', '   ', '   ', '   ', '   ']
['         ', '         ', '         ', '         ', '   ']

EL AGENTE AVANZA HACIA LA CELDA [1, 1]

SE A DETECTADO ['A']
AGREGANDO REGISTROS
EN LA CELDA: [1, 2]
EN LA CELDA: [2, 1]
Array: [[8, [0, 1]], [14, [2, 0]]]

['   AG    ', '   SEG    ', '   ', '   ', '   ']
['   AG    ', '   AG    ', '   ', ['A'], '   ']
['   SEG   ', '   ', ['A'], '   ', '   ', '   ']
['         ', '         ', '         ', '         ', '   ']

EL AGENTE AVANZA HACIA LA CELDA [0, 1]

Array: [[12, [0, 2]], [14, [2, 0]]]

['   AG    ', '   AG    ', '   SEG    ', '   ', '   ']
['   AG    ', '   AG    ', '   ', ['A'], '   ']
['   SEG   ', '   ', ['A'], '   ', '   ', '   ']
['         ', '         ', '         ', '         ', '   ']

EL AGENTE AVANZA HACIA LA CELDA [0, 2]
```

```

SE A DETECTADO ['A']
AGREGANDO REGISTROS
EN LA CELDA: [0, 3]
EN LA CELDA: [1, 2]
HOYO EN LA CELDA [1, 2]
Array: [[14, [2, 0]]]
Array: [[14, [2, 0]]]

[' AG ', ' AG ', ' AG ', ['A']]
[' AG ', ' AG ', ' |H| ', ' ']
[' SEG ', ['A'], ' ', ' ']
[' ', ' ', ' ', ' ']

```

EL AGENTE AVANZA HACIA LA CELDA [2, 0]

```

Array: [[4, [3, 0]], [11, [2, 1]]]

[' AG ', ' AG ', ' AG ', ['A']]
[' AG ', ' AG ', ' |H| ', ' ']
[' AG ', ' SEG ', ' ', ' ']
[' SEG ', ' ', ' ', ' ']

```

EL AGENTE AVANZA HACIA LA CELDA [3, 0]

```

Oro encontrado en la celda [3, 0]
[' AG ', ' AG ', ' AG ', ['A']]
[' AG ', ' AG ', ' |H| ', ' ']
[' AG ', ' SEG ', ' ', ' ']
[' AG-G ', ' ', ' ', ' ']

```