



Universidad de Granada

decsai.ugr.es

Inteligencia Artificial

Seminario 2

Agentes Reactivos / Deliberativos



DECSAI

**Departamento de Ciencias de la
Computación e Inteligencia Artificial**

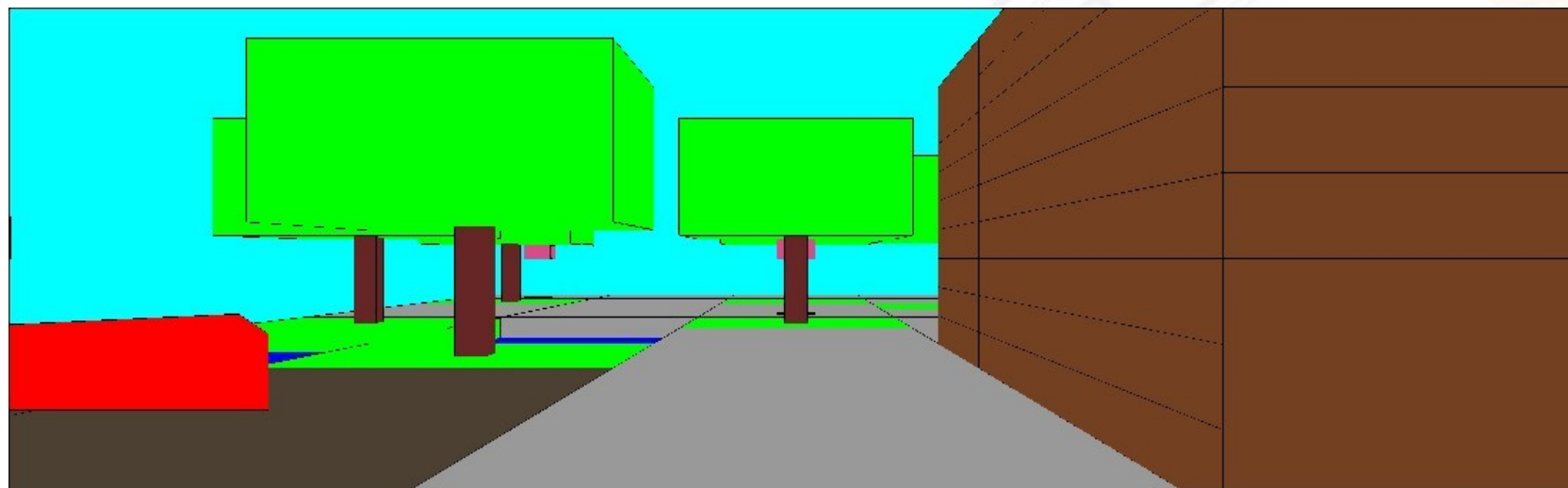
- **Introducción**
- **Los extraños mundos de BelKan**
- **Objetivos**
- **Software**
- **Método de evaluación y entrega de prácticas**



- **Introducción**
- Los extraños mundos de Belkan
- Objetivos
- Software
- Método de evaluación y entrega de prácticas



- Diseñar e implementar un agente reactivo y deliberativo, capaz de percibir el ambiente y actuar considerando una representación de las consecuencias de sus acciones y siguiendo un proceso de búsqueda.



- En esta práctica se diseñará e implementará un agente reactivo y deliberativo basado en los ejemplos del libro *Stuart Russell, Peter Norvig, “Inteligencia Artificial: Un enfoque Moderno”, Prentice Hall, Segunda Edición, 2004.*
- El simulador que utilizaremos fue inicialmente desarrollado por el profesor Tsung-Che Chiang de la NTNU (Norwegian University of Science and Technology, Trondheim), pero la versión sobre la que se va a trabajar ha sido desarrollada por los profesores de la asignatura.

- Originalmente, el simulador estaba orientado a experimentar con comportamientos en aspiradoras inteligentes.

En su versión más simple, una aspiradora inteligente presenta un comportamiento **reactivo** puro: busca suciedad, la limpia, se mueve, detecta suciedad, la limpia, se mueve, y continúa con este ciclo hasta que se cumple alguna condición de parada.



- Otras versiones más sofisticadas permiten al robot recordar (mediante el uso de representaciones icónicas como mapas), lo cual permite que el aparato ahorre energía y sea más eficiente en su trabajo.
- Finalmente, las aspiradoras más elaboradas pueden, además de todo lo anterior, planificar su trabajo de modo que se pueda limpiar la suciedad en el menor tiempo posible y de la forma más eficiente. Estas últimas pueden ser catalogadas como **agentes deliberativos**.



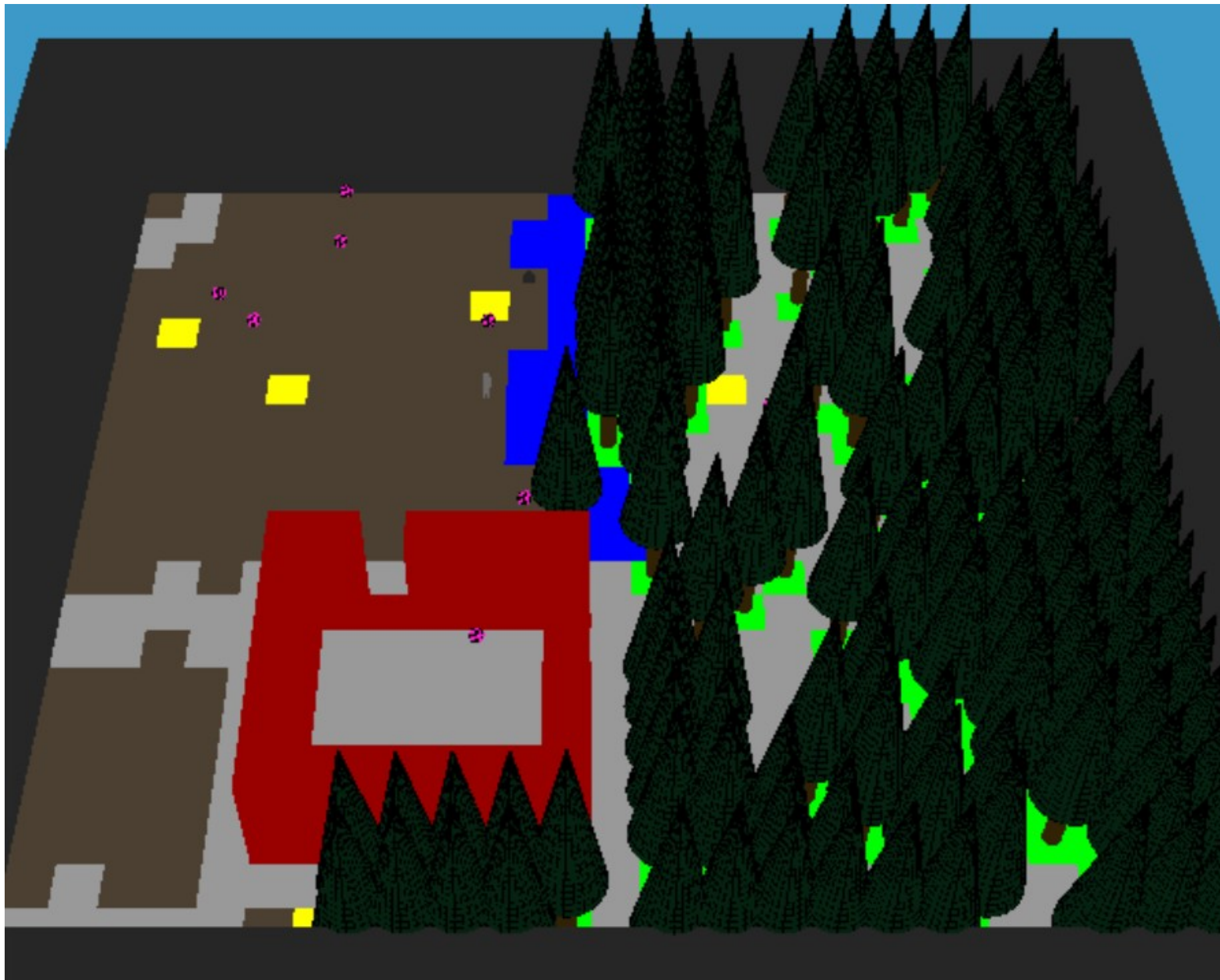
Esta práctica cubre los siguientes objetivos docentes:

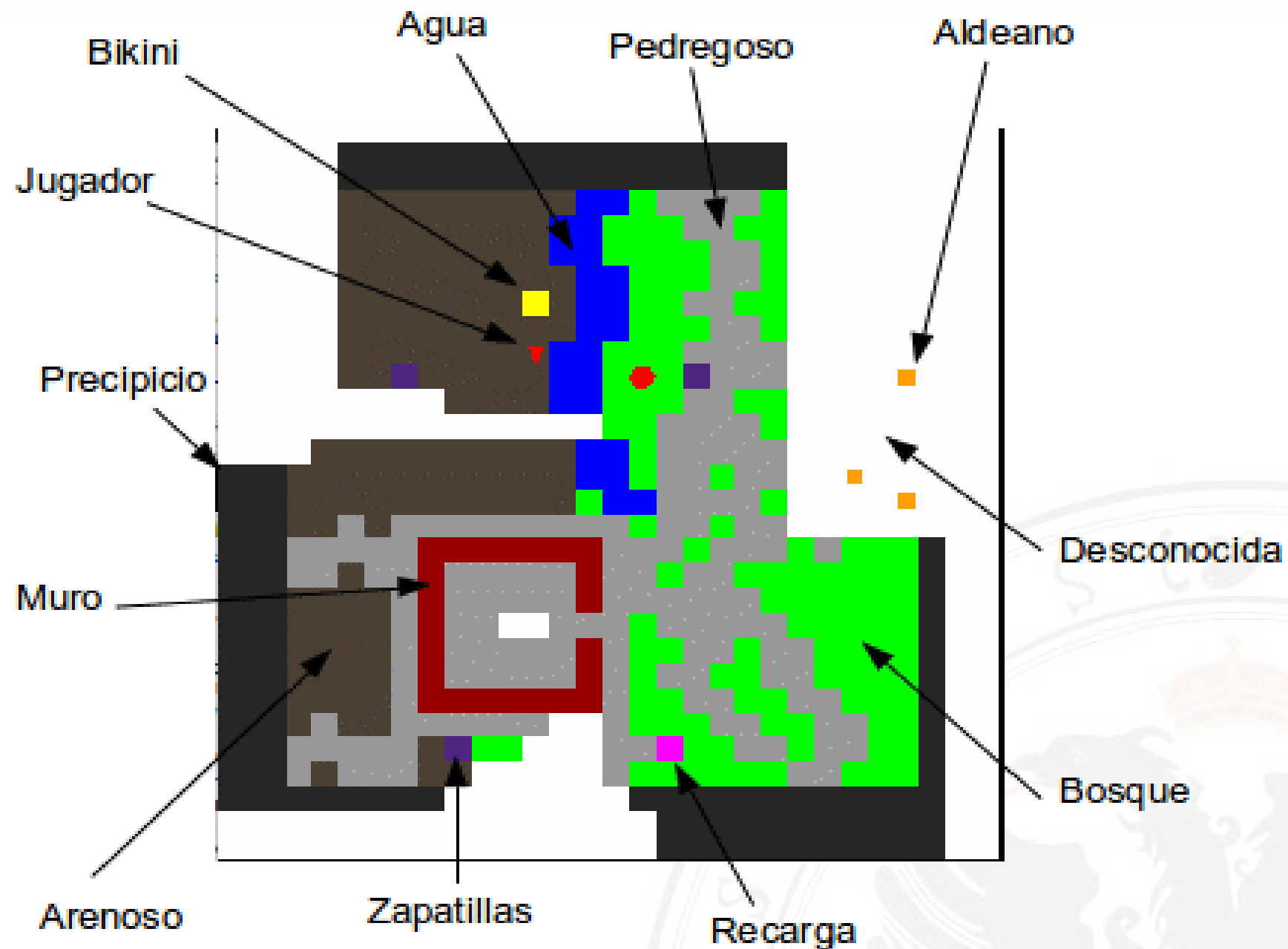
- Conocer la representación de problemas basados en estados (estado inicial, objetivo y espacio de búsqueda) para ser resueltos con técnicas computacionales.
- Entender que la resolución de problemas en IA implica definir una representación del problema y un proceso de búsqueda de la solución.
- Analizar las características de un problema dado y determinar si es susceptible de ser resuelto mediante técnicas de búsqueda. Decidir en base a criterios racionales la técnica más apropiada para resolverlo y saber aplicarla.
- Entender el concepto de heurística y analizar las repercusiones en la eficiencia en tiempo y espacio de los algoritmos de búsqueda.
- Conocer distintas aplicaciones reales de la IA. Explorar y analizar soluciones actuales basadas en técnicas de IA.

- Introducción
- **Los extraños mundos de Belkan**
- Objetivos
- Software
- Método de evaluación y entrega de prácticas



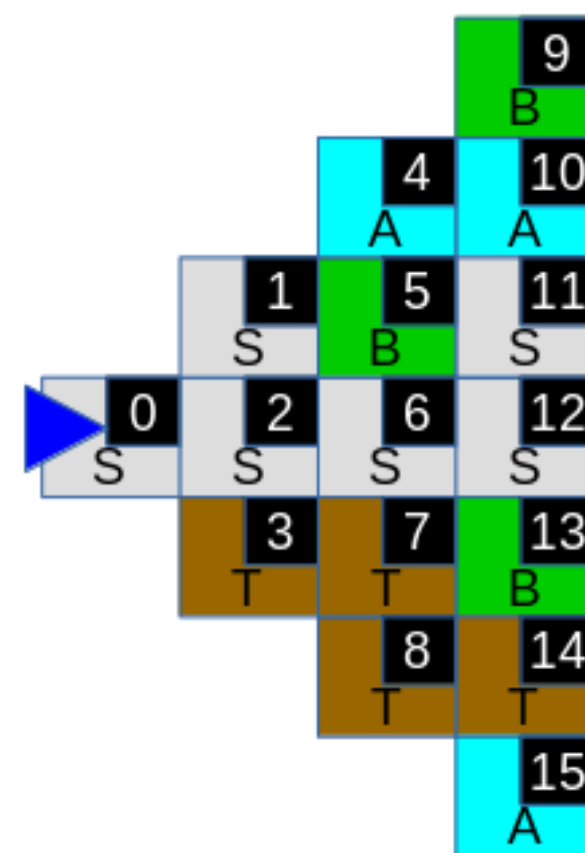
Los extraños mundos de BelKan





- Nuestro personaje no sabe volar, con lo cual si cae en un precipicio morirá.
- Tampoco puede atravesar, muros, ni otros personajes, y por tanto chocará contra ellos.

- Sensor de choque (colision)
- Sensor de vida (reset)
- Sensores de posición (posF, posC, sentido)
- Sensor de destino (destinoF, destinoC)
- Sensor de carga de batería (bateria)
- Sensor de nivel (nivel)
- Sensor de tiempo consumido (tiempo)



TERRENO / SUPERFICIE

Nuestro personaje puede realizar varias acciones distintas durante el juego:

- *actFORWARD*: le permite avanzar a la siguiente casilla del mapa siguiendo su orientación actual. Para que la operación se finalice con éxito es necesario que la casilla de destino sea transitable para nuestro personaje.
- *actTURN_L*: le permite mantenerse en la misma casilla y girar a la izquierda 90° teniendo en cuenta su orientación.
- *actTURN_R*: le permite mantenerse en la misma casilla y girar a la derecha 90° teniendo en cuenta su orientación.
- *actIDLE*: pues como su nombre indica, no hace nada.

- Cada acción realizada por el agente tiene un coste en tiempo de simulación y en consumo de batería.
- En cuanto al tiempo de simulación, todas las acciones consumen un instante independientemente de la acción que se realice y del terreno donde se encuentre el jugador.

- En cuanto al consumo de batería decir que actIDLE consume 0 de batería y que el consumo de este recurso de las acciones actTURN_L, actTURN_R y actFORWARD depende del tipo de terreno asociado a la casilla donde se inició dicha acción. En la siguiente tabla se muestran dichos valores de consumo:

Tipo de Casilla	Gasto Normal Batería	Gasto Reducido Batería
'A'	100	10 (con Bikini)
'B'	50	5 (con Zapatillas)
'T'	2	2
Resto de Casillas	1	1

- Introducción
- Los extraños mundos de Belkan
- **Objetivo**
- Software
- Método de evaluación y entrega de prácticas



- El objetivo de esta práctica es dotar de un comportamiento inteligente a nuestro personaje usando un agente reactivo/deliberativo para definir las habilidades que le permitan dentro del juego según el nivel seleccionado.
- Para que sea más fácil resolver esta práctica, se han diseñado dos niveles con dificultad incremental:
 - NIVEL 1: Agente deliberativo
 - NIVEL 2: Agente reactivo/deliberativo

Nivel 1: Agente deliberativo

- Su **objetivo** es crear y llevar a cabo un plan de movimientos en el mapa (*agente deliberativo*) para llegar con seguridad desde el origen al destino.
- ¡No debe morir ni chocarse con muros! Para ello es obligatorio implementar alguno de los algoritmos vistos en clase.

Nivel 1: Agente deliberativo

- Nuestro personaje se encuentra en un mundo inmutable y completamente conocido.
- Nuestro personaje aparecerá sobre un mundo de BelKan concreto conociendo su posición a través de los sensores **posF** y **posC** y orientación sobre el mapa a través del sensor **sentido**.
- El objetivo es construir un camino que le permita llegar hasta la posición de destino (marcada en los sensores **destinoF** y **destinoC**).
- Como ejemplo el Nivel 1 Comportamiento 1 está ya implementado usando la búsqueda en profundidad sobre grafos para que el estudiante lo use como ejemplo.

Nivel 2: Agente reactivo/deliberativo

- En este nivel el agente no conoce el mundo y debe explorarlo para conocerlo. El agente conoce su posición y orientación, también conoce el objetivo, pero no conoce el resto de elementos del mundo.
- Además en el mundo existe otros agentes llamados aldeanos. Un aldeano puede moverse continuamente, puede quedarse quieto, etc. Los aldeanos pueden detectarse gracias al sensor de superficie.
- Para ir de un punto a otro del mapa puede que nuestro plan inicial no funcione correctamente ya que nos crucemos en el camino con un aldeano que nos entorpezca => deberemos integrar comportamientos reactivo y deliberativo.

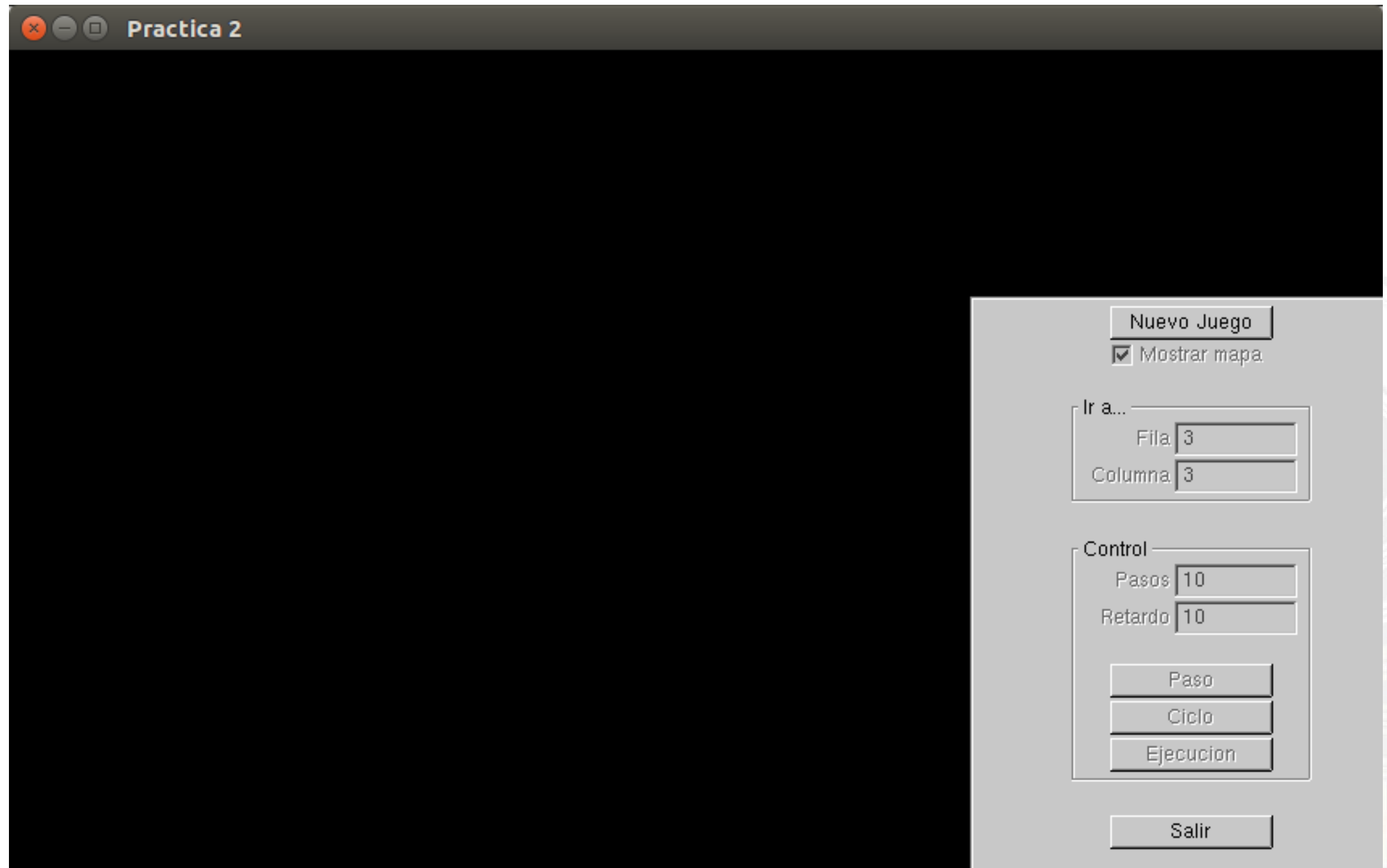
- Para optar por este nivel es necesario tener hecho el nivel anterior.
- En este nivel cada vez que lleguemos al destino, éste cambiará de sitio. Es decir, deberemos tratar de conseguir llegar a la mayor cantidad de destinos posible. Para este nivel también se cuenta con un número de 3000 acciones como máximo, una batería con 3000 de capacidad o 300" totales. La simulación termina cuando se consume alguno de esos recursos).
- También es importante recordar que el jugador no debe morir.

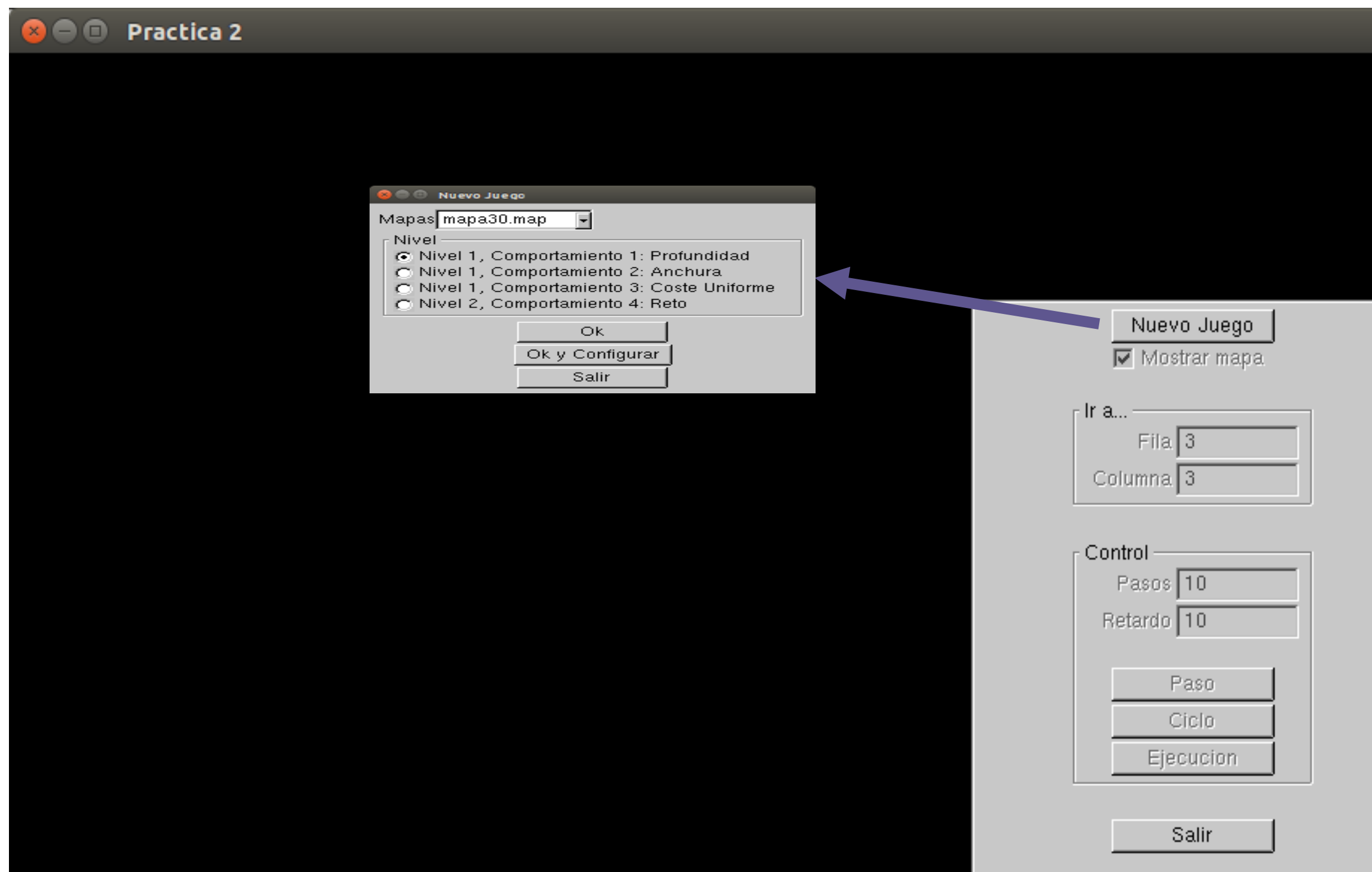
- Introducción
- Los extraños mundos de Belkan
- Objetivo
- **Software**
- Método de evaluación y entrega de prácticas

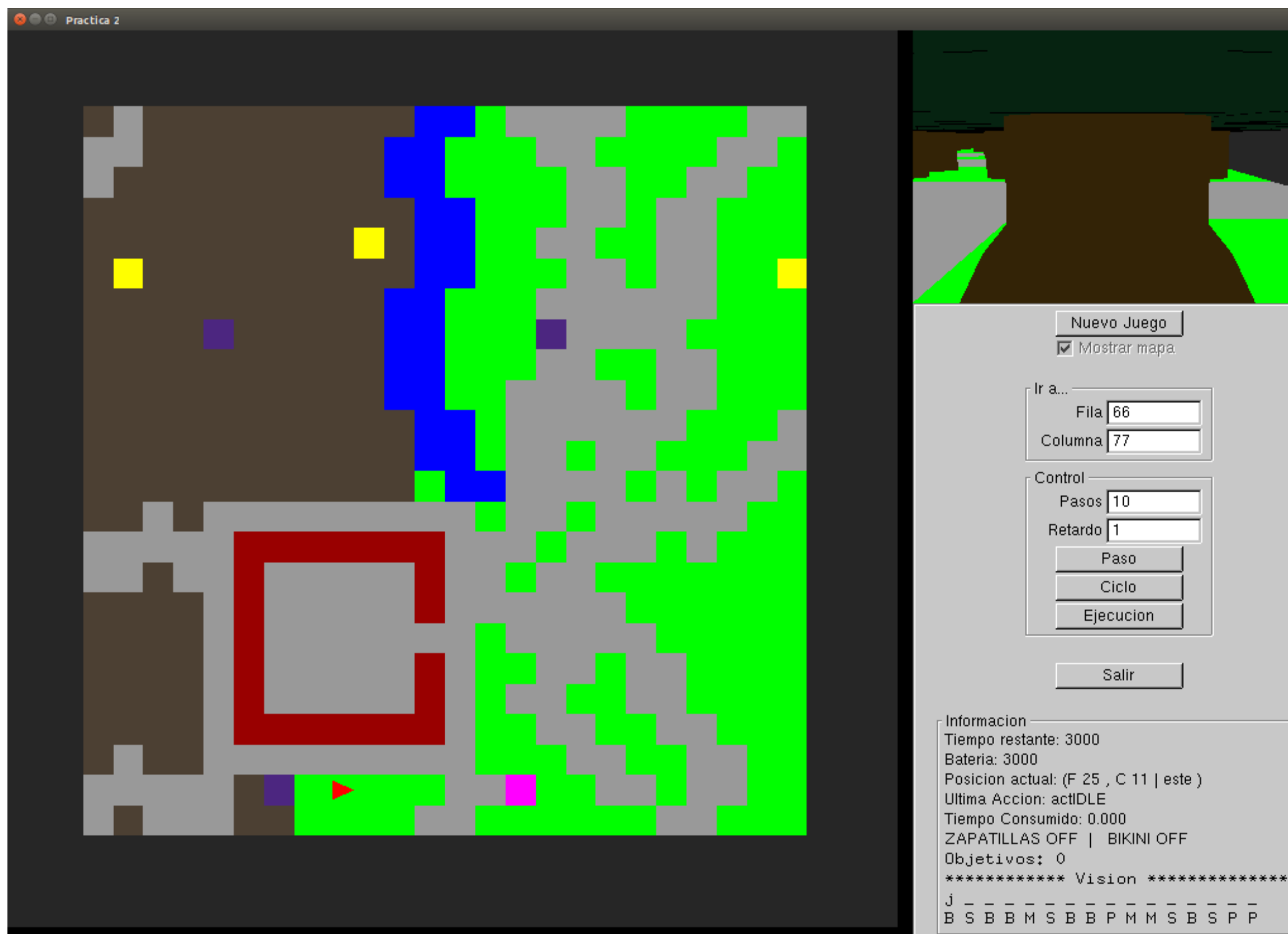


- Se proporcionan versiones del software para el sistema operativo Linux y para el sistema operativo Windows. Dicho software se puede encontrar en el apartado de “Material de la Asignatura” dentro de la plataforma docente de DECSAI.
- Existen dos ficheros ejecutables: Belkan y BelkanSG. El primero corresponde al simulador con interfaz gráfica, mientras que el segundo es un simulador en modo *batch* sin interfaz.
- Todos los detalles y explicación de la instalación, uso y detalles las variables necesarias para su desarrollo se encuentra en el guion de prácticas asociado a esta presentación.

`./Belkan <semilla>`







• Sistema *batch*:

`./BelkanSG mapas/mapas100.map 1 3 94 64 15 15 7 7`

- fichero de mapa
- semilla para inicializar el generador de números aleatorios
- Nivel (1, 2, 3 o 4)
- fila origen
- columna origen
- Orientación (0=norte, 1=este, 2=sur, 3=oeste)
- pares de (fila, columna) destino

En caso de quedarse sin destinos, los elegirá al azar.

Al finalizar la ejecución nos el tiempo consumido, la cantidad de batería con la que terminó la simulación, el número de colisiones que hemos sufrido (por obstáculos u otros agentes), la cantidad de muertes por caídas (al agua o a algún precipicio) y la cantidad de destinos alcanzados.

```
Tiempo Consumido: 0.170342  
Nivel Final de Bateria: 0  
Colisiones: 0  
Muertes: 0  
Objetivos encontrados: 6
```

- Introducción
- Los extraños mundos de Belkan
- Objetivo
- Software
- **Método de evaluación y entrega de prácticas**



- Se pide desarrollar un programa (modificando el código de los ficheros del simulador ‘jugador.cpp’ y ‘jugador.hpp’) con el comportamiento requerido para el agente.
- Estos dos ficheros deberán entregarse mediante la plataforma web de la asignatura, en un fichero ZIP (de nombre “practica2.zip” que no contenga carpetas ni sub-carpetas.
- El archivo ZIP deberá contener sólo el código fuente de estos dos ficheros con la solución del alumno así como un fichero de documentación en formato PDF que describa el comportamiento implementado con un máximo de 5 páginas.

- A partir del comportamiento entregado por el alumno en los ficheros antes mencionados y del nivel elegido por el mismo, se realizarán varias simulaciones con diferentes mapas (de diferente tamaño, pero con un máximo de 100x100) y con diferentes destinos. El simulador no debe “colgarse” en ningún momento y debe, por supuesto, compilar sin problemas.
- Se valorará:
 - Si se implementa tan sólo la búsqueda en anchura del Nivel 1 correctamente se obtendrán 3 puntos.
 - Si se implementa además correctamente la búsqueda con coste uniforme se obtendrán dos puntos adicionales.
 - Si se opta por el Nivel 2, esto implicará haber completado el Nivel 1 completamente y además realizar la tarea 2. En este caso, la nota máxima alcanzable será de 10. El nivel de la nota estará en función de destinos alcanzados (por tanto el objetivo de este nivel es alcanzar el máximo número de objetivos con la cantidad de acciones fijadas, la carga de la batería y el tiempo máximo preestablecido).

- La nota final se calculará de la siguiente manera:

$$\text{Nota final} = \text{Nota Practica} * \text{Defensa}$$

donde “Defensa” es un valor en $[0,1]$ indicando en que medida el alumno ha sabido defender los comportamientos implementados en su agente en el proceso de defensa.

- Si se opta por sólo búsqueda en anchura, la nota máxima alcanzable será de 5.
Si se opta por todo el nivel 1, la nota máxima alcanzable será de 5.
- Si se opta por el Nivel 2, la nota máxima alcanzable será de 10.

Grupo de prácticas	Fecha límite entrega	Fecha defensa
TODOS	3 de Mayo hasta las 23:00 horas	Semana del 4 de Mayo

Esta práctica es **INDIVIDUAL**

En el caso de detectar prácticas copiadas, los involucrados (tanto el que se copió como el que se ha dejado copiar) tendrán suspensa la asignatura.

