

# Introducción al curso resolución de problemas con algoritmos de búsqueda

Stalin Muñoz Gutiérrez

Centro de Ciencias de la Complejidad  
Universidad Nacional Autónoma de México (UNAM)

# Contenido

- Abstracción de un problema como un grafo de estados-acciones.

# Contenido

- Abstracción de un problema como un grafo de estados-acciones.
- Criterios de selección de algoritmos.

# Contenido

- Abstracción de un problema como un grafo de estados-acciones.
- Criterios de selección de algoritmos.
- Algoritmos de búsqueda ciega.

# Contenido

- Abstracción de un problema como un grafo de estados-acciones.
- Criterios de selección de algoritmos.
- Algoritmos de búsqueda ciega.
- Algoritmos de búsqueda informada.

# Contenido

- Abstracción de un problema como un grafo de estados-acciones.
- Criterios de selección de algoritmos.
- Algoritmos de búsqueda ciega.
- Algoritmos de búsqueda informada.
- Algoritmos metaheurísticos.

# Contenido

- Abstracción de un problema como un grafo de estados-acciones.
- Criterios de selección de algoritmos.
- Algoritmos de búsqueda ciega.
- Algoritmos de búsqueda informada.
- Algoritmos metaheurísticos.
- Implementación de programas en lenguaje Python.

# Definición de problema





# Definición de problema



## Definición del problema

- Comenzamos por identificar los estados relevantes del problema.

# Definición del problema

- Comenzamos por identificar los estados relevantes del problema.
- Después identificamos las acciones del agente.

## Definición del problema

- Comenzamos por identificar los estados relevantes del problema.
- Después identificamos las acciones del agente.
- Conectaremos los estados con otros estados a través de acciones del agente, formando un **grafo de estados-acciones**.

[illegible]

## Grafo de estados-acciones



## Grafo de estados-acciones







## Otros ejemplos

- 1 El rompecabezas del ocho o del quince.

## Otros ejemplos

- 1 El rompecabezas del ocho o del quince.
- 2 El cubo de rubik.

## Otros ejemplos

- 1 El rompecabezas del ocho o del quince.
- 2 El cubo de rubik.
- 3 Planificación automática para un robot.

# Complejidad e Inteligencia Artificial

## Complejidad Computacional

En Inteligencia Artificial los problemas a resolver tienen la característica de que son **computacionalmente complejos**.

Esto quiere decir que no se conocen algoritmos que puedan resolver los problemas de manera eficiente.

Los problemas dejan de ser materia de la Inteligencia Artificial cuando algoritmos eficientes se desarrollan.

## Análisis asintótico de algoritmos

- En el análisis asintótico de algoritmos nos interesa poder acotar los requerimientos de un programa ya sea memoria o tiempo con una función.
- Analizamos el peor de los escenarios para un algoritmo, es decir, cuando este consume el máximo de recursos computacionales.

Decimos que un algoritmo que consume un recurso a razón de una función  $T(n)$  con el parámetro  $n$ , es  $O(f(n))$  si existen  $n_0, k \in N$  y  $f(n)$  tal que:

$$T(n) \leq kf(n) \text{ para toda } n > n_0.$$

# Análisis asintótico de algoritmos

- Algoritmo para encontrar el valor máximo en una lista de números no ordenados.

```
1  $m \leftarrow -\infty$   
2 for  $i \in \text{lista}$   
   1 if  $i > m$  :  $m \leftarrow i$   
3 return  $m$ 
```

En este caso decimos que el algoritmo es  $O(n)$ .