

# Proyecto de Introducción a la Inteligencia Artificial con Matlab

## Generación 36 de pre-becarios del PROTECO

El proyecto consiste en la creación de una aplicación en **MATLAB** que permita encontrar la ruta más óptima de una estación  $x$  a otra estación  $y$  de la red de transporte público metro de la CDMX utilizando como fundamento teórico la **teoría de agentes** y los **problemas de búsqueda**.

El proyecto consta de 2 partes: un trabajo de investigación, una aplicación con su respectiva descripción .

Cualquier aclaración puede ser consultada personalmente o por medio de correo electrónico con Nicky García Fierros (instructor titular del curso intersemestral de IA), Rodrigo Castillo Alcántara (instructor titular del curso de selección de pre becarios de Matlab). A cualquiera de las siguientes direcciones:

- [nick.garc.96@gmail.com](mailto:nick.garc.96@gmail.com)
- [rodrigo.castillo.proteco@gmail.com](mailto:rodrigo.castillo.proteco@gmail.com)

### Sobre la entrega

- La realización del proyecto será en equipos de 2 a 4 personas, siendo libre la elección de compañeros.
- Deberá crearse un repositorio en GitHub segmentado con la sección de código y la sección de documentación. Además, debe incluirse un Readme corto pero descriptivo acerca de la aplicación.
- El último commit deberá realizarse como máximo el domingo 10 de junio de 2018 a las 23:59
- Cualquier contratiempo deberá ser reportado con prontitud a las direcciones de correo electrónico de los instructores del curso.

# Especificaciones para la aplicación

## Descripción del problema

El sistema de transporte colectivo **Metro** de la **CDMX** es un modo de transporte eficiente y rápido si sabes navegar bien en él. Por este propósito, has decidido aplicar los conocimientos adquiridos en la investigación acerca de teoría de agentes y problemas de búsqueda para desarrollar un prototipo de aplicación en Matlab capaz de decirte cuál es la ruta más óptima para ir de una estación a otra.

## Características necesarias de la aplicación

La vista de la aplicación puede ser sencilla o muy compleja. Deberá permitir al usuario indicar o seleccionar la estación de origen y la estación de destino. Tras finalizar su selección, la ruta encontrada deberá ser mostrada ante el usuario (incluyendo una lista de todas las estaciones y transbordos necesarios), así como el costo total de tomar dicha ruta en unidades de tiempo. La función de costo es definida por los desarrolladores de la aplicación, pero debe existir alguna relación en tiempo.

Una vez finalizado el cálculo de la ruta, se deberá mostrar el tiempo que tardó la aplicación en calcularla.

Se tienen total libertad creativa en cuanto a la vista y funcionalidad adicional a la descrita en el párrafo anterior. Puede manejarse desde el Command Window, pero se considerarán puntos extra si se realiza sobre una interfaz gráfica desarrollada en Matlab. La única restricción es que **EL AGENTE, MODELO, FUNCIÓN DE COSTO, ESTRATEGIA DE BÚSQUEDA Y HEURISTICA (EN CASO DE NECESITARLA) DEBERÁN SER PROGRAMADAS POR USTEDES.**

## Acerca del código

- Todo el código debe estar segmentado en funciones de Matlab que realicen tareas en específico. Tener grandes bloques de código sin coherencia baja puntos.
- Todas las funciones deben estar comentadas con una descripción de la tarea que realiza.
- Todo el código debe estar vectorizado a fin de hacerlo lo más elegante posible. Más información en: [https://la.mathworks.com/help/matlab/matlab\\_prog/vectorization.html](https://la.mathworks.com/help/matlab/matlab_prog/vectorization.html)

# Especificaciones para el reporte descriptivo de la aplicación

Tras haber desarrollado los temas anteriores, deberán aplicarse los conceptos al desarrollo de la aplicación. Esto debe redactarse en un LiveScript de Matlab con todas las imágenes, ecuaciones y gráficas necesarias. El contenido de este reporte es el siguiente:

- Planteamiento del problema
  - Actuadores
  - Acciones
  - Espacio de estados (Modelo)
  - Función de meta
  - Función de heurística (En caso de elegir alguna estrategia de búsqueda informada **fundamentar porqué se trata de una heurística admisible**)
- El algoritmo elegido para el problema de búsqueda
  - Demostración de la complejidad temporal
  - Demostración de la complejidad espacial
  - Eficiencia sobre otras estrategias
  - Pseudocódigo
  - Ejemplo de aplicación del pseudocódigo (puede recurrirse al uso de la aplicación ya desarrollada para mostrar el funcionamiento)
- Conclusiones
  - Deberá incluir cualquier dificultad encontrada (conceptual, de implementación, temporal, etc.)
  - Concluir las hipótesis planteadas en la introducción
  - Experiencia personal **por integrante del equipo**. (Lo que aprendí, lo que me gustó, lo que no me gustó).
  - Sugerencias para el desarrollo del curso intersemestral.
- Fuentes consultadas
  - Se recomienda bastante que la información se adquiera de fuentes confiables. A fin de cuentas, **toda esta información se puede adquirir de un solo libro**; buscar en internet podría resultar en una pérdida fatal de tiempo o en conceptos erróneos.
  - Se recomienda consultar fuentes en inglés.

# Especificaciones para el trabajo de investigación

El reporte escrito deberá redactarse completamente en LaTeX y deberá incluir lo siguiente:

- Introducción
  - Para la introducción, se deberá incluir una breve descripción general de lo que consiste el proyecto, los conceptos fundamentales necesarios para entender el trabajo escrito, las preguntas sobre las que se fundamenta la investigación y sus correspondientes hipótesis.
- Desarrollo (mínimo 10 cuartillas y máximo 40 cuartillas)
  - Esta sección deberá contener el desarrollo de los siguientes temas a investigar.
- ¿Qué es la inteligencia artificial?
  - Enfoques de la inteligencia artificial
    - Actuar como humano: Enfoque cognitivo
    - Pensar como humano: Enfoque por las *'leyes del pensamiento'*
    - Actuar de manera racional: Enfoque por agentes racionales
- Fundaciones de la inteligencia artificial:
  - Filosofía
  - Matemáticas
  - Economía
  - Neurociencia
  - Psicología
  - Ingeniería en computación
  - Teoría de control y cibernética
  - Lingüística
- Historia de la inteligencia artificial:
  - Gestión de la inteligencia artificial (1943-1955)
  - Nacimiento de la inteligencia artificial (1956)
  - Entusiasmo temprano, grandes expectativas (1952-1969)
  - Sistemas Expertos (1969-1979)
  - La IA se vuelve industria (1980-presente)
  - El regreso de las redes neuronales (1986-presente)
  - La IA adopta el método científico (1987-presente)
  - Surgimiento de los agentes inteligentes (1995-presente)
  - Grandes bancos de datos (Big Data) (2001-presente)
- Estado actual de la Inteligencia Artificial
  - ¿Qué puede hacer la IA hoy en día?
  - ¿Qué no puede hacer la IA hoy en día?
- Agentes Inteligentes
  - Agentes inteligentes
    - Agentes y entorno
    - Buen comportamiento: El concepto de racionalidad
    - Omnisciencia, aprendizaje, y autonomía
      - Racionalidad vs omnisciencia
      - Racionalidad vs perfección

- Dependencia de conocimiento previo: Falta de autonomía
    - Entorno de tareas
  - Especificación del entorno de tareas:
    - Rendimiento
    - Entorno
    - Actuadores
    - Sensores
  - Propiedades del entorno de tareas: **(Incluir al menos 1 ejemplo por cada propiedad)**
    - Visibilidad completa
    - Visibilidad parcial
    - Único agente vs Multi-agente
    - Determinístico vs estocástico
    - Episódico vs secuencial
    - Estático vs dinámico
    - Discreto vs continuo
    - Conocido vs desconocido
  - Estructura de agentes:
    - Agentes reflexivos simples
    - Agentes reflexivos basados en modelos
    - Agentes basados en metas
    - Agentes basados en utilidades
    - Agentes que aprenden
- Resolución de problemas mediante búsqueda
  - Agentes que resuelven problemas
    - Problemas bien definidos y soluciones
    - Formulación de problemas
  - Travelling Salesman Problem (TSP)
  - Búsqueda de soluciones
  - Parámetros de rendimiento de solución de problemas
    - Completitud
    - ¿Qué tan óptimo es?
    - Complejidad espacial
    - Complejidad temporal
  - Estrategias de búsqueda ciega
    - Búsqueda primero por amplitud (BFS)
    - Búsqueda primero por profundidad (DFS)
    - Búsqueda de costo uniforme (UCS)
  - Estrategias de búsqueda heurística
    - Búsqueda voraz (Greedy best-first search)
    - A Estrella (A\*)
  - Funciones heurísticas
    - Heurísticas admisibles
    - Aprendizaje de heurísticas por experiencia