

# Tema 1

Pedro O. Pérez M., PhD.

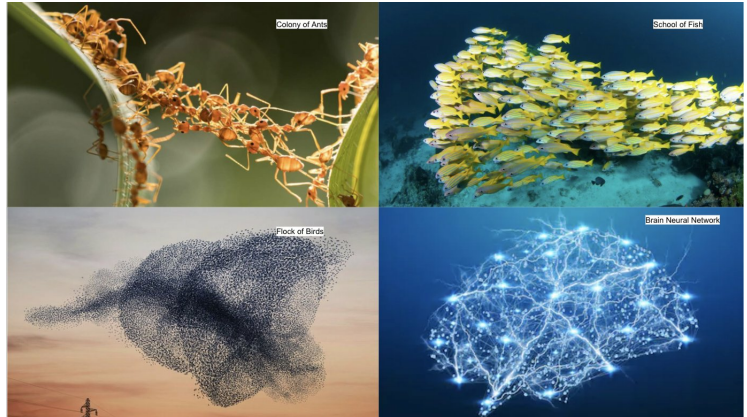
Modelación de sistemas multiagentes con gráficas computacionales  
Tecnológico de Monterrey

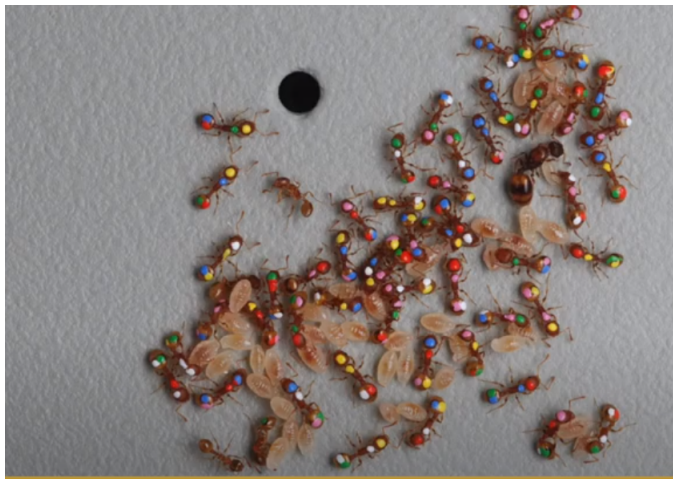
*pperezm@tec.mx*

10-2022

# Sistemas complejos

- Los sistemas complejos se componen de muchos elementos interconectados, que trabajan individualmente, pero producen un comportamiento general del sistema global.





<https://www.youtube.com/watch?v=C7E2rd-GmuI>

- Este fenómeno, denominado **comportamiento emergente**, es una consecuencia directa de los comportamientos individuales dentro del sistema y sus interacciones entre sí.
- Un comportamiento emergente que es impredecible o desconocido para el observador al comienzo de la simulación se considera un efecto caótico.

Los sistemas complejos se estudian de dos maneras:

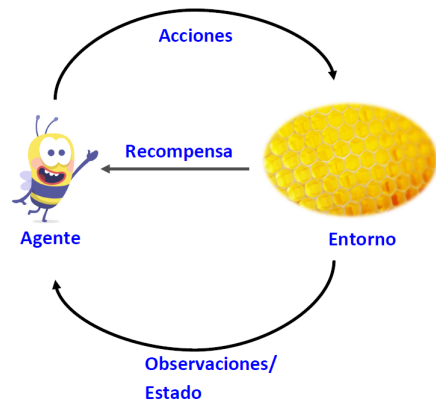
- Como un sistema colectivo.
- Como una colección de individuos que interactúan entre sí para producir un comportamiento general.

- El **comportamiento individual dinámico** se puede estudiar utilizando fórmulas matemáticas como ecuaciones diferenciales o actividades basadas en el tiempo. Sin embargo, el uso de ecuaciones matemáticas a menudo restringe los modelos a ciertos niveles de complejidad y recopilación de datos.
- En **sistemas complejos**, las interacciones individuales locales provocan cualidades emergentes del sistema en niveles superiores, lo que permite que la emergencia sea una consecuencia de lo que sucede dentro de estos micro niveles.

¿Qué otros sistema complejos  
conocen?



- Los **elementos individuales** existen en múltiples niveles dentro del sistema de jerarquías (pudiendo actuar de manera independiente).
- Estos **elementos** pueden actuar como **representantes de un solo individuo ejecutante o como una colección de individuos**, como grupo de múltiples individuos.
- Cada elemento **evalúa su comportamiento con base a su sistema de recompensa** y se adapta para funcionar mejor en las condiciones actuales del sistema.
- El sistema de recompensa está determinada por una medida de desempeño, donde los individuos usan receptores para leer señales y funciones para evaluar estos desempeños.



- La **adaptabilidad de los elementos** es una característica única que poseen los sistemas complejos. Su **capacidad para hacer frente a los cambios y su supervivencia** hace que los sistemas sean extremadamente robustos y favorables para la inspiración en aplicaciones de ingeniería.
- La adaptabilidad del sistema está muy influenciada por la presencia de ruido y atractores en el sistema.
- Los atractores son puntos ambientes que hacen que los elementos se desvíen de sus caminos ideales de comportamiento, posiblemente cuando los sistemas comienzan a mostrar comportamiento caóticos.

- **El sistema es parte de un sistema complejo más grande.** Los sistemas están conectados a otros sistemas como una jerarquía o utilizando ramas de entrada o salida.
- **Los sistemas son sistemas abiertos.** Los sistemas están interactuando con otros sistemas continuamente sin límites.
- **Los sistemas están cambiando dinámicamente.** Los sistemas están interactuando con otros sistemas continuamente sin límites.
- **Muestra un comportamiento emergente.** El comportamiento emergente se puede estudiar a niveles macro. Por ejemplo, las colonias de insectos logran sus objetivos más rápidamente al trabajar en convivencia entre hormigas individuales.
- **Los individuos se adaptan.** Según el entorno cambiante y los recursos del sistema disponibles, los individuos adapta su comportamiento para sobrevivir en determinadas condiciones.
- **Los individuos son egoístas.** Trabajan para su propio beneficio utilizando información local.

# Actividad colaborativa (30 minutos)

En parejas,

- Revisa cualquiera de los modelos que se encuentran en la siguiente liga:  
<http://ccl.northwestern.edu/netlogo/models/>
- Comenten que es lo que representa el modelo. ¿Cómo se comportan los individuos? ¿Cuál es el comportamiento emergente? ¿Qué problema s resuelve?

# Principales tendencias de la computación

# Principales tendencias de la computación

La historia de la computación ha estado marcada por cinco tendencias importantes y continuas:

- **Ubicuidad:** Por ubicuidad, nos referimos a la continua reducción en el costo de la capacidad informática. Esto ha permitido introducir capacidad de procesamiento en lugares y dispositivos donde hasta ahora hubiera sido antieconómico, y quizás incluso inimaginable.
- **Interconexión:** Los primeros sistemas informáticos eran entidades aisladas que se comunicaban solo con sus operadores humanos. Ahora los sistemas informáticos actuales suelen estar interconectados.
- **Inteligencia:** Actualmente nuestra capacidad de automatizar y delegar más y más tareas a las computadoras ha crecido de manera constante.

- **Delegación:** Cada día es más el control que le damos a los sistemas informáticos.
- **Orientación humana:** La quinta y última tendencia es el alejamiento constante de las visiones de la interacción humano-computadora orientadas a las máquinas hacia conceptos y metáforas que reflejan más de cerca la forma en que nosotros mismos entendemos el mundo.

Las tendencias hacia el aumento de la delegación y la inteligencia implican la necesidad de construir sistemas informáticos que puedan actuar eficazmente en nuestro nombre. Esto, a su vez, implica dos capacidades. El primero es la capacidad de los sistemas para operar de forma independiente, sin nuestra intervención directa. El segundo es la necesidad de que los sistemas informáticos puedan actuar de tal manera que representen nuestros mejores intereses mientras interactúan con otros seres humanos o sistemas.



- Juntas, estas tendencias han llevado al surgimiento de un nuevo campo en la informática: los sistemas multiagente.
- Pero, ¿qué es un agente? Un agente es un sistema informático que puede realizar acciones independientes en nombre de su usuario o propietario.
- Entonces, Un sistema multiagentes es aquel que consta de varios agentes, que interactúan entre sí, normalmente mediante el intercambio de mensajes a través de alguna infraestructura de red informática.

En el caso más general, los agentes de un sistema multiagente representarán o actuarán en nombre de usuarios o propietarios con objetivos y motivaciones muy diferentes. Para interactuar con éxito, estos agentes requerirán la capacidad de *cooperar, coordinar y negociar entre sí*, de la misma manera que nosotros cooperamos, coordinamos y negociamos con otras personas en nuestra vida cotidiana.

# Diseño

El primer problema es el del diseño del agente y el segundo problema es el del diseño de la sociedad. Los dos problemas no son ortogonales; por ejemplo, para construir una sociedad de agentes que trabajen juntos de manera eficaz, puede ser útil dar a los miembros de la sociedad modelos de los demás agentes que la componen.

# Niveles Macro/Micro

- Los investigadores en sistemas de agentes múltiples pueden estar preocupados principalmente por los sistemas de ingeniería, pero esta no es de ninguna manera su única preocupación.
- La IA se ha centrado en gran medida en los problemas de inteligencia de las personas.
- Pero seguramente una gran parte de lo que nos hace únicos como especie es nuestra capacidad social. No solo podemos comunicarnos entre nosotros en idiomas de alto nivel, podemos cooperar, coordinar y negociar entre nosotros.

# Habilidad social

- ¿Cómo puede surgir la cooperación en sociedades de agentes interesados?
- ¿Cómo pueden los agentes interesados reconocer cuándo sus creencias, metas o acciones entran en conflicto, y cómo pueden llegar a acuerdos entre ellos sobre asuntos de interés propio, sin recurrir al conflicto?
- ¿Cómo pueden los agentes autónomos coordinar sus actividades para alcanzar metas de manera cooperativa?
- ¿Qué tipo de lenguajes comunes pueden utilizar los agentes para comunicar sus creencias y aspiraciones, tanto a las personas como a otros agentes?
- ¿Cómo pueden los agentes contruidos por diferentes individuos u organizaciones que utilizan diferentes plataformas de hardware o software estar seguros de que, cuando se comunican con respecto a algún concepto, comparten la misma interpretación de este concepto?



Algunos ejemplos...

Debido a una falla inesperada del sistema, una sonda espacial que se acerca a Saturno pierde contacto con su equipo de tierra con base en la Tierra y se desorienta. En lugar de simplemente desaparecer en el vacío, la sonda reconoce que ha habido una falla clave del sistema, diagnostica y aísla la falla, y se reorienta correctamente para hacer contacto con su personal de tierra.

Un sistema clave de control de tráfico aéreo en el aeropuerto principal de Ruritania falla repentinamente, dejando a los vuelos en las cercanías del aeropuerto sin soporte de control de tráfico aéreo. Afortunadamente, los sistemas autónomos de control del tráfico aéreo en los aeropuertos cercanos reconocen la falla de sus pares y cooperan para rastrear y hacer frente a todos los vuelos afectados. La situación potencialmente desastrosa transcurre sin incidentes.

Después del invierno más frío y húmedo que se haya registrado en el Reino Unido, necesitas desesperadamente unas vacaciones de última hora en un lugar cálido y seco. Después de especificar tus requisitos a su asistente digital, éste conversa con varios sitios web diferentes, que venden servicios como vuelos, habitaciones de hotel y alquiler de automóviles. Después de una dura negociación en tu nombre con una variedad de sitios, tu asistente te presenta un paquete de vacaciones.

# Sistemas multiagentes

El campo de los sistemas multiagente es altamente interdisciplinario: se inspira en áreas tan diversas como la economía, la filosofía, la lógica, la ecología y las ciencias sociales. No debería sorprendernos que, por lo tanto, existan muchas visiones diferentes de lo que trata el “proyecto de sistemas multiagente”.

# Agentes como paradigma de la ingeniería de software

# Agentes como paradigma de la ingeniería de software

Los ingenieros de software han obtenido una comprensión cada vez mejor de las características de la complejidad del software. Ahora se reconoce ampliamente que la interacción es probablemente la característica más importante del software complejo. Las arquitecturas de software que contienen muchos componentes que interactúan dinámicamente, cada uno con su propio hilo de control, y que participan en protocolos complejos y coordinados, son típicamente órdenes de magnitud más complejas de diseñar de manera correcta y eficiente que aquellas que simplemente calculan una función de alguna entrada a través de una sola hilo de control.



# Computación “interesada”

En primer lugar, ha habido un aumento espectacular del interés en el estudio y la aplicación de los mecanismos económicos en la informática. Por ejemplo, las subastas son un tipo de mecanismo económico bien conocido, utilizado para la asignación de recursos, que ha alcanzado una prominencia particular en la informática. Existen varias razones para este rápido crecimiento del interés, pero quizás lo más fundamental es que se reconoce cada vez más que una comprensión verdaderamente profunda de muchos (quizás la mayoría) de los sistemas distribuidos y en red solo puede llegar después de reconocer que tienen las características de los sistemas económicos.

# Agentes como herramienta para comprender las sociedades humanas

Los sistemas de multiagentes proporcionan una herramienta interesante y novedosa para simular sociedades, que puede ayudar a arrojar algo de luz sobre varios tipos de procesos sociales.

¿No son solo sistemas  
distribuidos/concurrentes?

# ¿No son solo sistemas distribuidos/concurrentes?

- La comunidad de sistemas concurrentes ha estado investigando durante varias décadas las propiedades de los sistemas que contienen múltiples componentes que interactúan y ha estado desarrollando teorías, lenguajes de programación y herramientas para explicar, modelar y desarrollar tales sistemas.
- Los sistemas multiagente son, por definición, una subclase de sistemas concurrentes, y hay algunos en la comunidad de sistemas distribuidos que cuestionan si los sistemas multiagente son lo suficientemente diferentes de los sistemas distribuidos/concurrentes “estándar” como para merecer un estudio por separado.

En los sistemas de agentes múltiples, sin embargo, hay dos giros importantes en la historia de los sistemas concurrentes.

- Debido a que se supone que los agentes son autónomos generalmente se asume que las estructuras de sincronización y coordinación en un sistema de múltiples agentes no están programadas en el momento del diseño como es el caso de los sistemas concurrentes/distribuidos estándar. Por tanto, necesitamos mecanismos que permitan a los agentes sincronizar y coordinar sus actividades en tiempo de ejecución.

- Los encuentros que ocurren entre elementos informáticos en un sistema multiagente son encuentros económicos, en el sentido de que son encuentros entre entidades interesadas. En un sistema clásico distribuido/concurrente, se supone implícitamente que todos los elementos informáticos comparten un objetivo común (hacer que el sistema general funcione correctamente). En los sistemas de agentes múltiples, se supone, en cambio, que los agentes se preocupan principalmente por su propio bienestar (aunque, por supuesto, actuarán en nombre de algún usuario/propietario).

¿No es solo inteligencia artificial?



# ¿No es solo inteligencia artificial?

El campo de los sistemas de agentes múltiples ha disfrutado de una relación íntima con el campo de la IA a lo largo de los años. De hecho, hasta hace relativamente poco tiempo era común referirse a los sistemas de agentes múltiples como un subcampo de la IA. Más recientemente, se ha convertido en una práctica cada vez más común definir el esfuerzo de la IA en sí misma como uno de construir un agente inteligente.

Hay varios puntos importantes que deben destacarse aquí:

- La IA se ha preocupado en gran medida (y, quizás, erróneamente) de los componentes de la inteligencia: la capacidad de aprender, planificar, comprender imágenes, etc. Por el contrario, el campo de los agentes se ocupa de las entidades que integran estos componentes, con el fin de proporcionar un sistema que sea capaz de tomar decisiones independientes.
- La IA clásica ha ignorado en gran medida los aspectos sociales de la agencia.

¿No es todo solo economía/teoría  
de juegos?

# ¿No es todo solo economía/teoría de juegos?

La teoría de juegos es una teoría matemática que estudia las interacciones entre agentes interesados. Es interesante notar que von Neumann, uno de los fundadores de la informática, también fue uno de los fundadores de la teoría de juegos. De hecho, al momento de escribir este artículo, la teoría de juegos parece ser la herramienta teórica predominante en uso para el análisis de sistemas multiagente.

¿No es todo solo ciencia social?

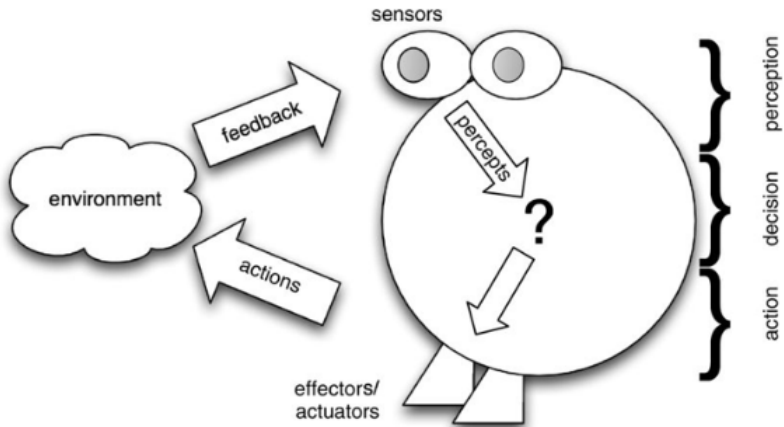
# ¿No es todo solo ciencia social?

Las ciencias sociales se preocupan principalmente por comprender el comportamiento de las sociedades humanas. Algunos científicos sociales está interesados en sistemas multiagente (computacionales) porque proporcionan una herramienta experimental con la que modelas sociedades humanas. Además, un enfoque obvio para el diseño de sistemas multiagente, que son sociedad artificiales, es observar cómo funcionan las sociedades humanas y tratar de construir el sistema multiagente de la misma manera.

¿Qué es un agente?

- Lamentablemente, no existe una definición universalmente aceptada del término agente y, de hecho, hay mucho debate y controversia en curso sobre este mismo tema. Esencialmente, si bien existe un consenso general de que la autonomía es fundamental para la noción de agencia, hay poco acuerdo más allá de esto.
- Sin embargo, en este curso nos basaremos en la definición propuesta por Wooldridge y Jennings: Un *agente* es un sistema informático que *está situado* en algún *entorno*, y que es capaz de actuar de *forma autónoma* en este entorno para cumplir sus objetivos delegados.





En la mayoría de los dominios de complejidad razonable, un agente no tendrá un control completo sobre su entorno. Tendrá, en el mejor de los casos, un control parcial, ya que puede influir en él. Desde el punto de vista del agente, esto significa que la misma acción realizada dos veces en circunstancias aparentemente idénticas puede parecer tener efectos completamente diferentes y, en particular, puede no tener el efecto deseado. Por lo tanto, los agentes en todos los entornos, excepto en los más triviales, deben estar preparados para la posibilidad de fallar.

¿Autónomo?

- Lo primero que hay que decir es que la autonomía es un espectro. En un extremo de este espectro se encuentran los seres humanos plenamente realizados, como tú y yo. Tenemos tanta libertad como cualquiera con respecto a nuestras creencias, nuestros objetivos y nuestras acciones. Por supuesto, no tenemos total libertad sobre creencias, metas y acciones. Pero, podemos elegir nuestras creencias, nuestras metas y nuestras acciones.
- En el otro extremo del espectro de la autonomía, podríamos pensar en un servicio de software (como un método público proporcionado por un objeto Java). Dichos servicios no son autónomos en ningún sentido significativo: simplemente hacen lo que se les dice.

El punto en el espectro de la autonomía que nos interesará en gran medida es una entidad en la que podemos delegar objetivos de alguna manera de alto nivel, y luego hacer que esta entidad decida por en sí misma cuál es la mejor manera de lograr sus objetivos. La entidad aquí no es completamente autónoma: no puede simplemente elegir qué objetivos lograr, excepto en la medida en que tales “subobjetivos” permitan lograr los objetivos delegados.

¿Racional?

- Un **agente racional** es aquel que hacer lo correcto. Es decir, para cada secuencia de percepciones, el agente devuelve la acción correcta.
- Para determinar qué es correcto, necesitamos definir una **medida de rendimiento** que evalúa cualquier **secuencia de estados del entorno**. Típicamente, la medida de rendimiento se diseña de acuerdo a lo que uno quiere que ocurra en el entorno, en lugar de lo que pensamos que debe hacer el agente.
- Pro ejemplo, en el caos de una aspiradora eléctrica, se puede considerar cada cuadro limpio en cada instante de tiempo. A su vez, se puede penalizar por la cantidad de electricidad consumida y el ruido generado cuando hay personas.

La racionalidad depende, entonces, de cuatro factores:

- ① La medida de rendimiento que define el criterio de éxito.
- ② El conocimiento a priori que el agente tiene sobre el entorno.
- ③ Las acciones que el agente puede realizar.
- ④ La percepción del agente en un momento dado.

Con esto en mente, podemos establecer una primera definición de agente racional: En cada posible secuencia de percepciones, un agente racional deberá emprender aquella **acción** que supuestamente **maximice su medida de rendimiento**, basándose en las **evidencias aportadas por la secuencia de percepciones** y en el **conocimiento que el agente mantiene almacenado**.



- Cuando diseñamos un agente, es importante considerar cuatro elementos:
  - La medida de rendimiento (**P**erformance measure).
  - El entorno (**E**nvironment).
  - Los **A**ctuadores del agente.
  - Los **S**ensores del agente.
- En el diseño de un agente, el primer paso debe ser siempre especificar el entorno de trabajo de la forma más completa.

<b>Tipo de agente</b>	<b>Medidas de rendimiento</b>	<b>Entorno</b>	<b>Actuadores</b>	<b>Sensores</b>
Sistema de análisis de imágenes de satélites	Categorización de imagen correcta	Conexión con el satélite en órbita	Determinar la categoría de una escena	Matriz de píxeles de colores
Robot para la selección de componentes	Porcentaje de componentes clasificados en los cubos correctos	Cinta transportadora con componentes, cubos	Brazos y manos articulados	Cámara, sensor
Controlador de una refinería	Maximizar la pureza, producción y seguridad	Refinería, operadores	Válvulas, bombas, calentadores, monitores	Temperatura, presión, sensores químicos.
Tutor de inglés	Maximizar la puntuación de los estudiantes en los exámenes	Conjunto de estudiantes, entidad evaluadora	Ejercicios, sugerencias, correcciones	Teclado, mouse

Ahora, nos enfocaremos en los entornos (o ambientes) y discutiremos las propiedades generales de los mismos.

- **Totalmente observable vs. parcialmente observable.** Un entorno es totalmente observable si los sensores detectan todos los aspectos que son relevantes para la toma de decisiones.
- **Determinista vs. estocástico.** Si el siguiente del entorno de trabajo está completamente determinado por el estado actual y por la acción que va a ejecutar en el entorno, se dice que el entorno es determinístico, de lo contrario, es estocástico.
- **Episódico vs secuencial.** En un entorno episódico, la experiencia del agente se divide en episodios atómicos. En cada episodio, el agente percibe una vez el entorno y realiza una sola acción. El siguiente episodio no depende de las acciones realizadas previamente. En entornos secuenciales, la decisión presente puede afectar decisiones futuras.

- **Estático vs. dinámico.** Si el entorno puede cambiar mientras el agente está deliberando (procesando), el entorno es dinámico. De lo contrario, es estático. Si el entorno no cambia con el paso del tiempo, pero el rendimiento del agente cambia, entonces se dice que el entorno es semidinámico.
- **Discreto vs. continuo.** La distinción entre continuo o discreto aplica al estado del entorno, es decir, la forma en la que se maneja el tiempo, las percepciones y acciones del agente.
- **Individual vs. multiagente.**
- **Competitivo vs. cooperativo.** Para entornos multiagentes; competitivo se refiere cuando un agente intenta maximizar su medida de rendimiento, provocando la minimización de la medida de rendimiento de otros agentes. Cuando un agente busca maximizar su medida de rendimiento y esto maximiza la medida de rendimiento de otros agentes el entorno se define como cooperativo.

- **Conocido vs desconocido.** En un entorno conocido, los resultados de las acciones (o las probabilidad de los resultado si el entorno es estocástico) están bien definidas y se conocer. Si el entorno es desconocido, el agente tiene que aprender cómo sus acciones afectan el entorno.

<b>Tipo de agente</b>	<b>Observable</b>	<b>Determinista</b>	<b>Episódico</b>	<b>Estático</b>	<b>Discreto</b>	<b>Agentes</b>
Sistema de análisis de imágenes de satélites	Totalmente	Determinista	Episódico	Semi	Continuo	Individual
Robot para la selección de componentes	Parcialmente	Estocático	Episódico	Dinámico	Continuo	Individual
Controlador de una refinería	Parcialmente	Estocático	Secuencial	Dinámico	Continuo	Individual
Tutor de inglés	Parcialmente	Estocático	Secuencial	Dinámico	Discreto	Multi

# Actividad colaborativa (30 minutos)

En parejas,

- Definan las características de los siguientes ambientes: crucigrama, partido de ajedrez con reloj, póquer, backgammon, taxi circulando.
- Seleccionen un entorno (o situación), determinen las propiedades generales que tiene.

# Ejemplos



Un sistema de control puede verse como un agente. Un ejemplo simple de tal sistema es un termostato. Los termostatos tienen un sensor para detectar la temperatura ambiente. Este sensor está integrado directamente en el entorno (es decir, la habitación) y produce como salida una de dos señales: una que indica que la temperatura es demasiado baja y otra que indica que la temperatura es correcta. Las acciones disponibles para el termostato son “encender calefacción” o “apagar calefacción”. La acción “encender calefacción” generalmente tendrá el efecto de elevar la temperatura de la habitación, pero esto no puede ser un efecto garantizado; si la puerta de la habitación está abierta, por ejemplo, es posible que encender la calefacción no tenga ningún efecto.

La mayoría de los “daemons” de software (como los procesos en segundo plano en el sistema operativo Unix), que monitorean un entorno de software y realizan acciones para modificarlo, pueden verse como agentes. Un ejemplo es el programa de X Windows *xbiff*. Esta utilidad monitorea continuamente el correo electrónico entrante de un usuario e indica a través de un ícono de GUI si tiene o no mensajes no leídos.

Los agentes son simplemente sistemas informáticos que son capaces de actuar de forma autónoma en algún entorno para cumplir sus objetivos de diseño. Un agente normalmente detectará su entorno (mediante sensores físicos en el caso de agentes situados en parte del mundo real, o mediante sensores de software en el caso de agentes de software) y tendrá disponible un repertorio de acciones que se pueden ejecutar para modificar el medio ambiente, que puede parecer responder de manera no determinista a la ejecución de estas acciones.

¿Qué características debería tener  
un agente inteligente?

La siguiente lista fue sugerida por Wooldridge y Jennings:

- **Proactividad.** Los agentes inteligentes pueden exhibir un comportamiento dirigido a objetivos al tomar la iniciativa para satisfacer sus objetivos de diseño.
- **Reactividad.** Los agentes inteligentes son capaces de percibir su entorno y responder de manera oportuna a los cambios que ocurren en él para satisfacer sus objetivos de diseño.
- **Habilidad social.** Los agentes inteligentes son capaces de interactuar con otros agentes (y posiblemente con humanos) para satisfacer sus objetivos de diseño.

¿Qué no es un agente?

Los objetos se definen como entidades computacionales que encapsulan algún estado, pueden realizar acciones o métodos en este estado y comunicarse mediante el paso de mensajes. Si bien existen similitudes obvias, también existen diferencias significativas entre agentes y objetos.

La visión tradicional de un objeto y nuestra visión de un agente tienen al menos tres distinciones:

- Los agentes encarnan una noción de autonomía más fuerte que los objetos y, en particular, deciden por sí mismos si realizan o no una acción a petición de otro agente. Esta distinción entre objetos y agentes se ha resumido muy bien en el siguiente lema: “Los objetos lo hacen gratis; los agentes lo hacen porque quieren”.
- Los agentes son capaces de tener un comportamiento flexible (reactivo, proactivo, social) y el modelo de objeto estándar no tiene nada que decir sobre este tipo de comportamiento.
- Un sistema de agentes múltiples es intrínsecamente multiproceso, en el sentido de que se supone que cada agente tiene al menos un hilo de control.



Los sistemas expertos fueron la tecnología de IA más importante de la década de 1980. Un sistema experto es aquel que es capaz de resolver problemas o dar consejos en algún dominio rico en conocimiento. Un ejemplo clásico de un sistema experto es MYCIN, que estaba destinado a ayudar a los médicos en el tratamiento de infecciones de la sangre en humanos. MYCIN funcionó mediante un proceso de interacción con un usuario para presentar al sistema una serie de hechos (representados simbólicamente), que el sistema luego utilizó para derivar alguna conclusión. MYCIN actuó en gran medida como consultor: no operó directamente en humanos ni en ningún otro entorno.

En resumen, las principales diferencias entre agentes y sistemas expertos son las siguientes:

- Los sistemas expertos “clásicos” son incorpóreos: no están acoplados a ningún entorno en el que actúan, sino que actúan a través de un usuario como “intermediario”.
- Los sistemas expertos generalmente no son capaces de un comportamiento reactivo y proactivo.
- Los sistemas expertos generalmente no están equipados con capacidad social, en el sentido de cooperación, coordinación y negociación.

# Tipos de agentes

Existen cuatro tipos de tipos básicas (diseño) para agentes:

- Agentes reactivos simples.
- Agentes reactivos basados en modelos.
- Agentes basados en objetivos.
- Agentes basados en utilidad.