# Penn CIS 5210 - Chapter 2

Jonathon Delemos - Dr. Chris Callison Burch

June 7, 2025

This course investigates algorithms to implement resource-limited knowledge-based agents which sense and act in the world. Topics include, search, machine learning, probabilistic reasoning, natural language processing, knowledge representation and logic. After a brief introduction to the language, programming assignments will be in Python. — Description of CIS 4210/5210 in course catalog

## 1 Chapter Two: Intelligent Agents

### 1.1 2.1 - Agents and Environments

**Agent** is anything that can be viewed as perceiving its environment through sensors and acting upon the environment through **actuators**. Agent behavior is mapped through a Mathematical function called **agent functions.** Internally, we look at the agent function as a **agent program.** The example in the book they use is vaccuum world.

**Rational Agent** is a mechanical agent that does the right thing at the right time. **Consequantiliasm** is the idea that the agent flows through a series of states. The sequence of states is determined to be desirable or not based off the **performance measure.**

### 1.2 2.2/2.3 - Good Behavior and Nature of Environments

**PEAS** - Performance, Environments, Acuators, Sensors. The word **stochastic** is used interchangeably with nondeterministic. A model is stochastic if it explicitly deals with probability. It's nondeterministic if the probabilites aren't included. I.e. "there's a chance it may rain tomorrow.

### 1.3 2.4 - The Structure of Agents

**Agent Architecture** – Agent = Architecture + Program. Architecture is the hardware. Program is the agent program. Agent programs will likely not keep an active history of all tabled actions.

This would result in exponential growth due to the Fundamental Counting Principle:

$$\sum_{t=1}^{T} |P|^t$$

When it comes to defining the learning agent, the most important distinction is between the **learning element**, which is responsible for making improvements, and the **performance element**. **Utility** is the term we use to describe the measured success of an agent. **Critic** is the part of the AI model which focuses on how an agent is doing, also providing feedback. **Problem Generator** is responsible for suggesting actions that could lead to learning. Some of these actions are initially suboptimal. These four components make up the learning agent.

## 1.4 2.4.7 - How the components of the agent work

**Atomic Representation** is looked at like a black box. A -¿ B, and it has no internal structure. Think of it like a circuit from A -¿ B. It's a road you cannot deviate from. **Factored representation** splits up each state into a fixed state of variables of attributes. Think of it like an array instead of a variables (Atomic). **Structured representation** can be equated to a relational database, where there are tables of objects and each object can roughly quantify what another object is. Think of it like objects and relationships. The difference between these three models is **expressivenes**. That is representation that is captured based on the model. If there is a one-to-one mapping between concepts and memory locations, we call that a localist representation.

## 1.5 Questions?

How could this be translated into a test?
Could we use an array of values that we update to dictate how decisions are made? Is that what we are doing?

## 1.6 Summary

In this chapter, we discusssed agents, agent functions (the input output from percepts), performance measures, rational agents, task environments, agent programs, simple reflex agents, model-based reflex agents, goal-based agents, and utility-based agents. All agents improve their performance through learning.