Philosophy 240
Symbolic Logic

Russell Marcus
Hamilton College

Fall 2013

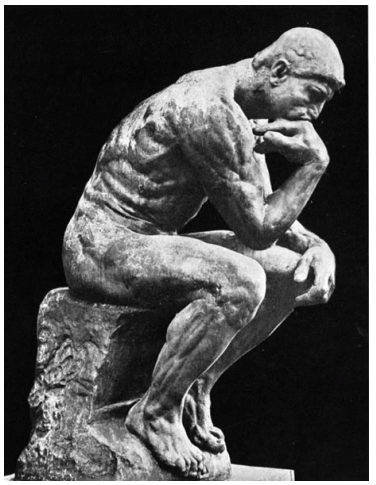
Class 13: Friday, September 27
Syntax, Semantics and the Chinese Room

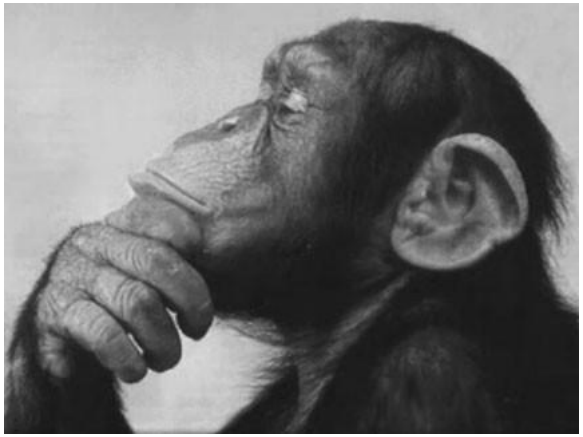
Outline

- Quick overview of the philosophy of mind
  - Three mainly-discredited theories
  - Functionalism as the dominant view
- The relation between functionalism and AI
  - AI: It is possible for artifacts (e.g. computers, robots, etc.) to have full-fledged mental states.
- Some concerns about functionalism and AI arising from considerations of logic.
  - 17 Slides

Three (Old) Theories of Mind

1. Dualism
   - Minds are non-physical substances (souls).
   - Main problem: interaction
2. Behaviorism
   - Mental states are behaviors, or dispositions to behave.
   - Main problem: mental states with no attached behaviors
     - chess player
3. Mind-Brain Identity Theory
   - Minds are brains.
   - Main problem: chauvinism





Functionalism

The Now-Dominant Theory of Mind

- Mental states are defined by their inputs (stimuli), outputs (resultant behaviors), and connections with other mental states.
- The mind is the software of the brain.
- Anything that behaves like something with a mind, and that has internal processes that map onto our internal processes, has a mind.
- Functionalism accommodates the dualist's emphasis on the connections among mental states, without the problem of interaction.
  - Functionalists are usually materialists, but need not be.
- Functionalism accommodates the behaviorist's emphasis on observable output without the problem of missing internal states.
- Functionalism accommodates the materialist's emphasis on the importance of the hardware without limiting mental states to creatures with our hardware.

Computer Scientists are Often Functionalists

"Saying Deep Blue doesn't really think about chess is like saying an airplane doesn't really fly because it doesn't flap its wings" (Drew McDermott, Computer Science, Yale University).

McCarthy's Thermostat

has three beliefs

1. that it is too cold
2. that it is too hot
3. that it is just right

Functionalism and Chauvinism

- Maybe ascribing beliefs to a thermostat is extravagant.
  - Mosquitos?
  - Chimps?
  - Dolphins?
- Functionalism nicely avoids the chauvinism of mind-brain identity theory.

Zombies
A Problem for Functionalism

- But not these kinds of zombies: philosophical zombies (brain-eating not required)
- Put this worry aside.

Strong AI

- Searle's argument opposes the claim, by defenders of artificial intelligence, that computers can think.
- Weak AI thesis: machines built to perform tasks that humans perform can give us some insight into the nature of our thought.
  - mostly uncontroversial
- Strong AI claim: computers (can) actually have minds.
  - Cheap calculators can now perform very complicated tasks, and quickly.
  - Machines are already able to do many tasks that once were inconceivable:
  - Proving mathematical theorems that require more computation than humans can perform.
  - Chess
- Better machines may approach or overtake human skill in other areas.
- All we need in order to have a mind is to simulate the behavior, along with some plausible internal causes of that behavior.

Syntax and Semantics

- Computers and their software work according to purely formal, syntactic manipulation.
- The syntax of a program or system of formal logic concerns its form, or shape.
- The semantics of a system or program concerns the meanings, or interpretations, of its terms.

Object Languages and Meta-Languages

- The object language is the language that we are studying.
- The meta-language is the language we use to study the object language.
- Rules for wffs
  - written in the meta-language
  - about how to construct an object language
- Rules for constructing truth tables
  - written in a meta-language
  - we use ‘1' and ‘0' which are not symbols of our object language.
  - Truth tables themselves are written in the meta-language.
- The syntax tells how the formulas are constructed.
- The semantics tells how to interpret the formulas.
- Inference rules and rules of equivalence are specified syntactically, too.

Separating Syntax from Semantics

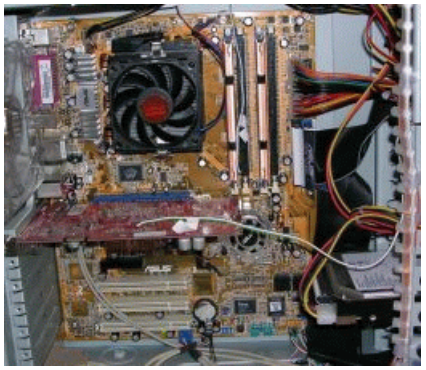
- We can treat our formal languages as completely uninterpreted, or topic-neutral.
- We can play with the symbols, according to the rules we specify, as if they were meaningless toys.
- We can interpret our languages variously, comparing interpretations in order to see the properties of the language itself clearly.
- Frege wanted a syntactic criterion for logical consequence.
  - To ensure that all deductions are secure
  - To ensure that we do not implicitly smuggle into our results unjustifiable interpretations
    - Non-Euclidean geometries
    - Infinitesimals

Frege on Syntax and Semantics

from Begriffsschrift

"So that nothing intuitive could intrude [into our concept of logical consequence] unnoticed, everything had to depend on the chain of inference being free of gaps. In striving to fulfil this requirement in the strictest way, I found an obstacle in the inadequacy of language: however cumbersome the expressions that arose, the more complicated the relations became, the less the precision was attained that my purpose demanded...The present *Begriffsschrift*...is intended to serve primarily to test in the most reliable way the validity of a chain of inference and to reveal every presupposition that tends to slip in unnoticed, so that its origin can be investigated."

How Computers Work



- Computers, in their most basic form, contain:
  - a complete list of possible states of the system (state table)
  - lists of possible inputs
  - lists of outputs computable from inputs and state table
  - all lists are specifiable syntactically
- Computer programs are algorithms, like cooking recipes.
  - An algorithm is just a list of instructions, a procedure.
  - Recipes generally just give simple, linear instructions.
  - An algorithm can also do different things depending on the state of the system executing the algorithm.
- Some algorithms contain conditional clauses
  - if the machine is in such-and-such a state, and receives such-and-so input, then it does this-and-that and moves into this other state.
- Computers merely follow algorithms.
- Every step of the algorithm can be specified syntactically, by its inscription.

Appearance and Reality

- When we play a video game, we see cars and people, and hear music.
- We interact with the machine on a semantic level.
- But, the computer is just processing syntax, crunching 0s and 1s.
- If strong AI and functionalism are right, then human behavior must be describable algorithmically as well, and representable in purely syntactic form, using a formal language like the one we use in logic.
- Despite appearances, we are just information processors (according to strong AI).

The Chinese Room

Imagine that a bunch of computer programmers have written a program that will enable a computer to simulate the understanding of Chinese. So, for example, if the computer is given a question in Chinese, it will match the question against its memory, or data base, and produce appropriate answers to the questions in Chinese. Suppose for the sake of argument that the computer's answers are as good as those of a native Chinese speaker. Now then, does the computer, on the basis of this, understand Chinese, does it literally understand Chinese in the way that Chinese speakers understand Chinese? Well, imagine that you are locked in a room, and in this room are several baskets full of Chinese symbols. Imagine that you (like me) do not understand a word of Chinese, but that you are given a rule book in English for manipulating these Chinese symbols. The rules specify the manipulations of the symbols purely formally, in terms of their syntax, not their semantics. So the rule might say: 'Take a squiggle-squiggle sign out of basket number one and put it next to a squoggle-squoggle sign from basket number two.' Now suppose that some other Chinese symbols are passed into the room, and that you are given further rules for passing back Chinese symbols out of the room. Suppose that unknown to you the symbols passed into the room are called 'questions' by the people outside the room, and the symbols you pass back out of the room are called 'answers to the questions.' Suppose, furthermore, that the programmers are so good at designing the programs and that you are so good at manipulating the symbols, that very soon your answers are indistinguishable from those of a native Chinese speaker. There you are locked in your room shuffling your Chinese symbols and passing out Chinese symbols in response to incoming Chinese symbols.

Searle's Claim

Any syntactic processor, completely describable in terms of formal processing, is *necessarily* not a mind.

Searle's Argument

1. Brains cause minds.
2. Syntax is not sufficient for semantics.
3. Computer programs are entirely defined by their formal, syntactic structure.
4. Minds have semantic contents.

C1: Computer programs are not sufficient for minds (i.e. computers can not think).

C2: The way that brains cause minds can not be by running a computer program.

C3-4: Anything that causes minds, including any artefact that we might make, must have causal powers at least equivalent to those of the brain.