Limits to Computation

Assignment 5

- Evaluation
  - Style (formal language, grammar, spelling, organization, bibliography) 15%
  - Research (depth, breadth) 45%
  - Critical evaluation of arguments summarized/facts presented (give your thoughts and opinions!) 40%
- 500 words is a suggestion, just to give an indication of how much work is expected
- Standard rules on *plagiarism* apply. Do not attempt to copy someone else's work without giving due credit, as this constitutes a SERIOUS academic offense.

DARPA Grand Challenge

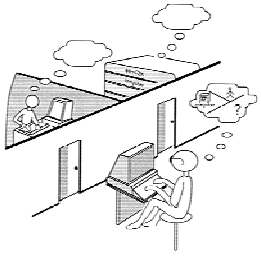
- Competition for driverless cars sponsored by DARPA
- Largest long distance competition for autonomous cars in the World
- Rule: Vehicle must be entirely autonomous, using only GPS and the information it detects with its sensors to navigate
- 2004 Grand Challenge
  - None of the teams succeeded the 227.2 km journey
- 2005 Grand Challenge
  - 5 vehicles completed challenge successfully
- 2007 Urban Challenge
  - must obey all traffic regulations while negotiating with other traffic and obstacles

Chatterbots

- Computer program designed to simulate an intelligent conversation with a human user
- Usually work by scanning for keywords within the input and pull a reply with the most matching keywords or the most similar wording pattern from a local database
- ELIZA:
  - 1966 computer program written by Joseph Weizenbaum
  - simulates/parodies (?) a therapist
  - http://www-ai.ijs.si/eliza/eliza.html
  - The ELIZA effect is the tendency to unconsciously assume computer behaviors are analogous to human behaviors, despite conscious knowledge to the contrary.
- ALICE:
  - One of the strongest current programs of its type
  - Winner of the Loebner prize in 2000, 2001, and 2004
  - http://www.alicebot.org/

Turing Test

- Test of a machine to perform human-like conversation
- Human judge engages in a natural language conversation with two other parties, one a human and the other a machine; if the judge cannot reliably tell which is which, then the machine is said to pass the test
- No computer has yet to pass the Turing test



Turing Test Validity

1. *Mechanical Objections*: A sufficiently fast machine with sufficiently large memory could be programmed with a large enough number of human questions and human responses to deliver a human answer to almost every question, and a vague random answer to the few questions not in its memory. This would simulate human response in a purely mechanical way. Psychologists have observed that most humans have a limited number of verbal responses.
2. *Data Processing Objection*: Machines process data bit by bit. Humans process data holistically. In this view, even if a machine appears human in every way, to treat it as human is to indulge in anthropomorphic thinking.
3. *Argument From Consciousness*: This argument, suggested by Professor Jefferson Lister states, "not until a machine can write a sonnet or compose a concerto because of thoughts and emotions felt, and not by the chance fall of symbols, could we agree that machine equals brain". Turing replies by saying that we have no way of knowing that any individual other than ourselves experiences emotions, and that therefore we should accept the test.
4. *Theological Objection*: This states that thinking is a function of man's immortal soul and therefore a machine could not think. Turing replies by saying that he sees no reason why it would not be possible for God to grant a computer a soul if He so wished.

Turing Test indication of Machine “Intelligence”?

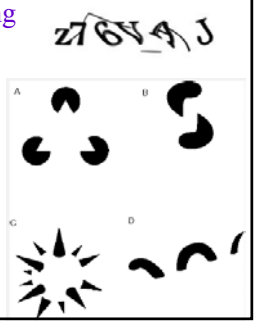
1. A machine passing the Turing test may be able to *simulate human conversational behaviour*, but this may be much weaker than true intelligence. The machine might just follow some cleverly devised rules. A common rebuttal in the AI community has been to ask, "How do we know humans don't just follow some cleverly devised rules?"
2. A machine may very well be intelligent without being able to chat like a human.
3. Many humans that we'd probably want to consider intelligent might fail this test (*e.g.,* the young or the illiterate). On the other hand, the intelligence of fellow humans is almost always tested exclusively based on their utterances.

Difficult problems

- Being able to hold a conversation with a human is thus a challenge for computers
- What other problems are difficult for computers?

(Visual) Perception

- Asking a user to type the distorted characters appearing in an image is a way to test that a human, rather than a computer, is performing the transaction.
- Observing optical illusions



Computers are "bad" at:

- Creativity
  - convince yourself by checking out computer generated poetry
- Doing anything without very detailed instructions

The Halting Problem

Given a program and an input to the program, determine if the program will eventually stop when it is given that input.

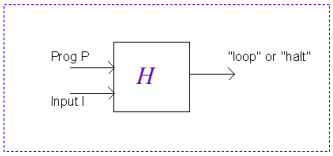
Example:

```
var x = prompt("Enter a number","")
x = parseInt(x)
while(x>5)
{
   document.writeln(x)
}
```

A human can give an answer for this program by inspecting it, but can a computer give an answer for all programs?

The Halting Problem

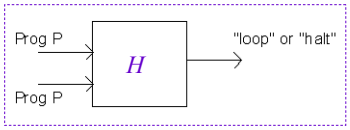
- Suppose there exists such a program *H* that given as input a description of a program *P* and an input *I*, returns "halt" if *P* halts on *I* or "loop" if it loops forever.



Note: there is nothing strange in a program taking a description of another program as input. When you open a Web page that has JavaScript in it, you give your JavaScript program as input to the Web Browser program.

The Halting Problem

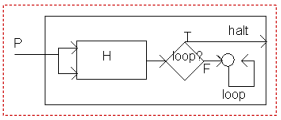
- Trick 1: You can give the description of your program *P* as both of the inputs to *H*. Now you are asking: does the program *P* halt when it is given *itself* as input?



The Halting Problem

- Trick 2: Make another program *K* that does the following:
  1. if *H* outputs "loop," then *K* halts
  2. if *H* outputs "halt," *K* loops forever.

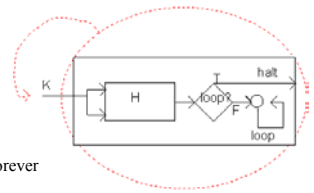
```
function K(P)
{
    if (H(P, P)=="loop") {
            return "halt" // halt
    } else {
            while(true) // loop forever
    }
}
```



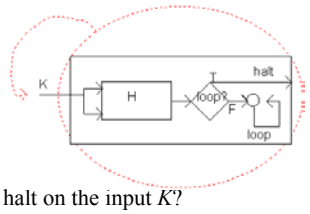
The Halting Problem

- Trick 3: Give the program *K* its own description *K* as input

```
function K(K)
{
    if (H(K, K)=="loop") {
            return "halt" // halt
    } else {
            while(true) // loop forever
    }
}
```



The Halting Problem



- Does *K* halt on the input *K*?
- If so, then *H* says that *K* halts on *K,* but then *K* loops and doesn't halt!
- If not, then *H* says that *K* loops on *K,* but then *K* halts!
- This makes *NO SENSE!!!* So, there cannot exist such a program *H* that can tell if a program halts on a given input.

The Halting Problem

- Thus there can be NO computer that can solve the Halting Problem
- There exist problems that are *impossible* for computers to solve!
- Proof is due to Alan Turing, 1936.

More on DARPA Grand Challenge

- http://www.pbs.org/wgbh/nova/darpa/see.html