Turing Machines

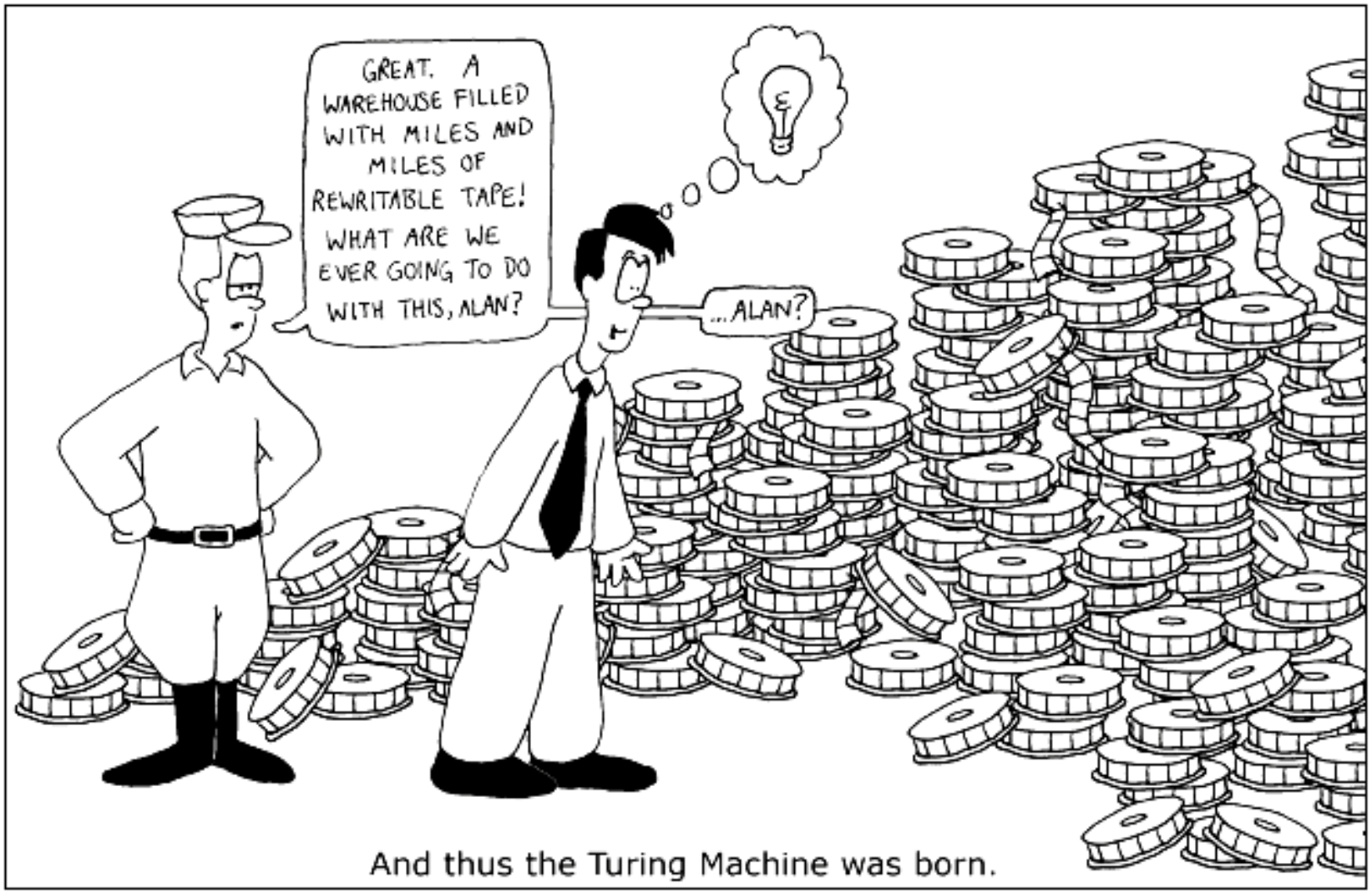
Instructor: Dmitri A. Gusev

Fall 2007

CS 210: Computing and Culture

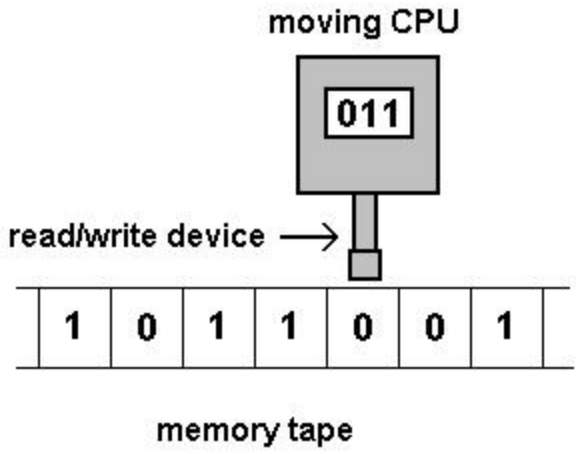
Lecture 4, September 24, 2007

- With the exception of its input/output mechanisms, the computer we've described last time was simply a finite-state machine connected to memory.


GREAT. A WAREHOUSE FILLED WITH MILES AND MILES OF REWRITABLE TAPE! WHAT ARE WE EVER GOING TO DO WITH THIS, ALAN?
...ALAN?
ε
And thus the Turing Machine was born.

Turing machine

- A *Turing machine* is a finite-state machine connected to an **infinitely long tape** using a moving **head** capable of reading and writing symbols on the tape



- How can we implement a traffic light controller using a Turing machine?

More Precisely…

- The infinite tape is divided into cells, and each cell contains a symbol from some **finite** alphabet that has 2 or more distinct symbols.
- The head can read and write symbols on the tape and **move** (the tape) left and right one and only **one cell at a time**
- The *action table* determines, based upon the current state of the machine and the input symbol just read, (a) which symbol the machine would write, (b) whether the head would then move right or left, and (c) which state the machine would assume next.
- The number of different states is always **finite**, and the *initial state* is known.

Computational Power

- Any computation that can be performed by any physical computing device can also be performed by a Turing machine, as long as the latter has sufficient time and memory.

Analog Computers

- “The very best analog computers have fewer than 30 bits of accuracy.”

Halting Problem

The *halting problem* is the problem of determining whether any program will eventually stop given a particular input. It is *unsolvable*. ☹

…We cannot build an algorithm able to decide whether **any** given mathematical (logical) statement is true or false. (Goedel's incompleteness theorem)

Pseudorandom Sequences and Quantum Computing

- *Pseudorandom* sequences produced by computers look random, but there is an underlying algorithm that generates them.
- “One could use a Geiger counter to generate truly random data sequences” – Does God play dice?