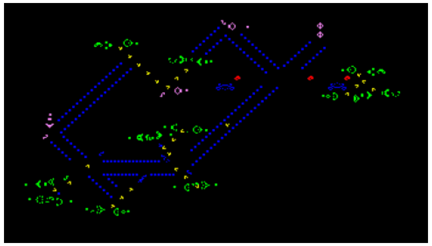
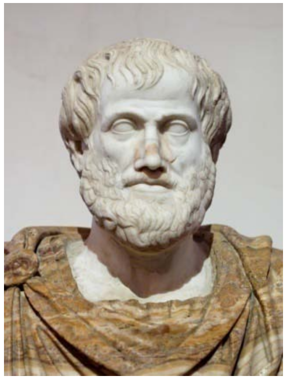
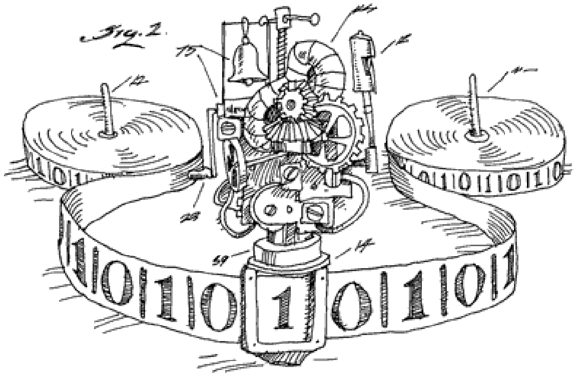
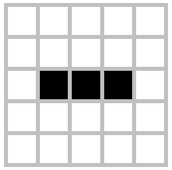
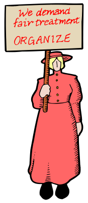





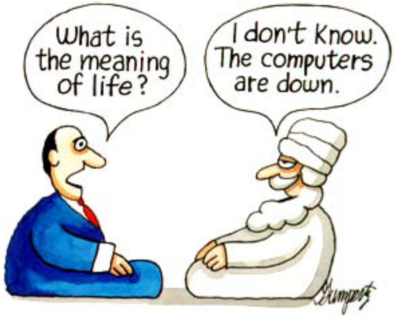


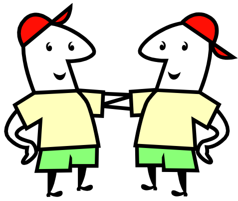
COMP2000 Quiz Notes

Computation and logic





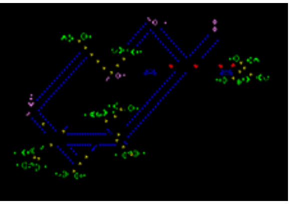





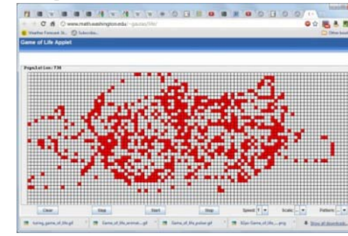
Computation: Turing machine

Current state	Reads	Writes	Moves	New state
Start_state	_	Y	Right	Halt
Start_state	0	_	Right	Start_state
Start_state	1	_	Right	Start_state
Start_state	_	N	Right	Halt

- Turing machine:
  - Can compute anything we know to be computable by other means (Church-Turing thesis)
  - Has an infinite tape with a read/write head, which starts with an input written on it. Has finitely many states in its description.
  - Simple rules: in a state, read a symbol then change a state (maybe), overwrite the symbol with another one (maybe) and move Left or Right
  - Invented by Alan Turing to show that some problems are not computable.

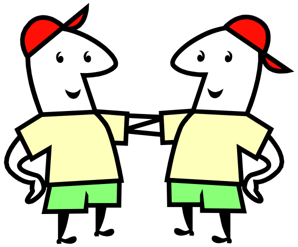


Computation: Game of Life



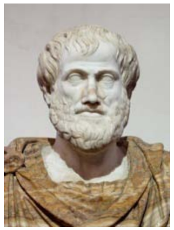
- Game of Life:
  - As powerful as a Turing machine: a Turing machine can simulate it moves, and it can simulate a Turing machine (non-trivial!)
  - Start with a board; some of the cells on the board are marked “live”. Every cell has 8 neighbours (above, below, sides and diagonal).
  - Rules:
    - If a live cell has fewer than 2, or more than 3 neighbours, it dies.
    - If a live cell has 2 or 3 neighbours, it stays alive.
    - If a dead cell has exactly 3 neighbours, it becomes live.
  - Many patterns: static, oscillating, moving

Logic: propositions

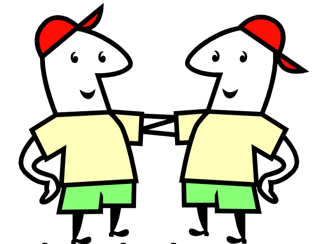


- *Propositions*: sentences that can be true or false
  - “It is Tuesday today” is true on a Tuesday and false on all other days.
  - “5 is a prime number” is true, “6 is prime” is false
  - Non-example: “ x is a prime number”. This is a predicate, it would become a proposition if x is set to a specific number. “Come here” is not a proposition either.
- *Logic connectives*: connect several propositions into a composite sentence that can be true or false, depending on the propositions. A sentence that is always true is called a *“tautology*”, always false *“contradiction*”.
- *Truth table*: a table listing all possible combinations of truth values of propositions, together with columns for composite statements. Can be used to find out when a composite statement is true and when it is false.
- Below is a truth table for the logic connectives we used:

A	B	not A	A and B	A or B	if A then B
True	*True*	False	True	True	True
True	*False*	False	False	True	False
False	*True*	True	False	True	True
False	*False*	True	False	False	True

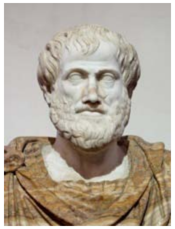



Logic: quantifiers

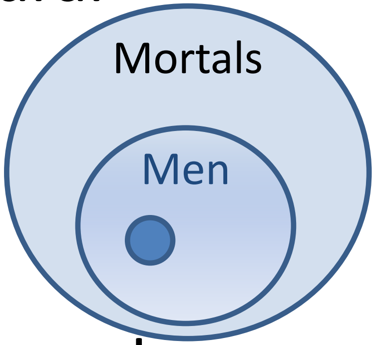


- A *predicate* is a “proposition with parameters”, which becomes a proposition if the parameters are set to a specific value.
  - “x is a prime number”. Prime(x)
  - “X loves Y”
- One can also make composite statements with predicates stating that for *every* or for *some* element of the kind the predicate is talking about something is true. The “every” ($\forall$) and “some” or “exists”, ($\exists$) are called quantifiers, universal and existential.
  - Every man is mortal
  - There exists a number that is prime (some numbers are prime)
  - It is foggy every day in St. John’s.
- Now, both propositional logic connectives and quantifiers can be combined to make composite statements of first-order logic.
  - Every man is an animal, and some animals have fur
  - Everybody loves somebody, and that somebody loves Elvis Presley.
    - $\forall$ X $\exists$ Y (X loves Y) and (Y loves Elvis)

Modus ponens

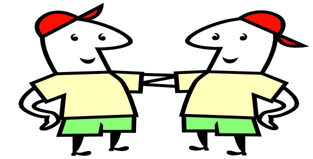


- Modus ponens is a rule used in proofs in natural deduction. Here is the general form:
  - If A then B
  - A holds
  - Therefore, B holds



- With universal quantifier, becomes universal modus ponens; second line substitutes specific value.
  - All man are mortal
  - Socrates is a man
  - Therefore, Socrates is mortal.
- Modus ponens allows solving some logic problems and puzzles faster than truth tables. But nobody knows if it is always better (there is a million dollar prize for that).


Negations



- To disprove a universal (“every”) statement, give a counterexample (“exists” statement).
  - It is foggy **every** day in St. John’s
    - No, on Monday it was sunny.
  - So “**not** (**every** day is foggy)” is **“exists** a day that was **not** foggy”.
  - But it’s definitely does not mean that every day is not foggy
  - “**not** (**exists** a day that is foggy)” is “**every** day is **not** foggy”.
- Similarly, negating an existential get universal of the “not”s:
  - There **exists** a natural number less than 1
    - No, **every** natural number is **not** less than 1.
- For logic connectives, notice that “if.. Then” is similar to “every”, and “and” is similar to “exists”:
  - **If** Socrates is a man, **then** Socrates is mortal
    - No, Socrates is a man, **and** he is immortal
  - Number 0.5 is a natural number **and** it is less than than 1
    - No, either 0.5 is **not** a natural number, **or** it is **not** less than 1.
  - Either today is Tuesday **or** today is Thursday
    - No, today is **not** Tuesday **and** today is **not** Thursday.

For the quiz:

- Be able to say what a transition of a Turing machine such as “in state A on reading 0 go to state B, write 1 and move right” does to the tape content.
- Be able to do a few generations of a game of life starting with a given pattern.
- There will be a few yes/no questions (e.g., is it true that everything we know to be computable is computable by a Turing machine?)

For the quiz

- Practice solving knight and knave puzzles and (simple) treasure hunts. Use truth tables and modus ponens, where appropriate.
- Make sure that you can recognize those fallacies we played with (which card to turn over? If I like blue triangle, what about yellow circle? Is Susan more likely to be a banker or a banker-activist?)
- Practice translating between logic and English, and identifying propositions and quantifiers in English sentences.
- Practice doing negations. See some new slides in Lecture 3, and last lecture.
- Email me if you have any questions! I will be around on Wednesday afternoon, if you'd like to stop by.