COMP 1002

Intro to Logic for Computer Scientists

Lecture 3
Admin stuff

• Labs: Wed 9am. First lab Jan 18\textsuperscript{th}.
  – CS-1019 (section 1, up to 60)
  – EN-1049 (section 2, up to 10)

  – If you do have a time conflict at 11am:
    • Come to EN-1049

  – If you do not have a time conflict at 11am:
    • Come to CS-1019
• On a mystical island, there are two kinds of people: knights and knaves. Knights always tell the truth. Knaves always lie.

• Puzzle 1: You meet two people on the island, Arnold and Bob. Arnold says “Either I am a knave, or Bob is a knight”. Is Arnold a knight or a knave? What about Bob?
Knights and knaves

Puzzle 1: You meet two people on the island, Arnold and Bob. Arnold says “Either I am a knave, or Bob is a knight”. Is Arnold a knight or a knave? What about Bob?

- A: Arnold is a knight
- B: Bob is a knight
- Formula: \( \neg A \vee B \) : “Either Arnold is a knave, or Bob is a knight”
- Want: scenarios where either both A is a knight and the formula is true, or A is a knave and the formula is false. Use “if and only if” notation: \( (\neg A \vee B) \leftrightarrow A \). True if both formulas have same value.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>( \neg A )</th>
<th>( \neg A \vee B )</th>
<th>( (\neg A \vee B) \leftrightarrow A )</th>
</tr>
</thead>
<tbody>
<tr>
<td>True</td>
<td>True</td>
<td>False</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td>False</td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>False</td>
<td>True</td>
<td>True</td>
<td>True</td>
<td>False</td>
</tr>
<tr>
<td>False</td>
<td>False</td>
<td>True</td>
<td>True</td>
<td>False</td>
</tr>
</tbody>
</table>
Special types of sentences

• A sentence that has a satisfying assignment is **satisfiable**.
  – Some row in the truth table ends with True.
  – Example: $B \rightarrow A$

• Sentence is a **contradiction**:
  – All assignments are falsifying.
  – All rows end with False.
  – Example: $A \land \neg A$

• Sentence is a **tautology**:
  – All assignments are satisfying
  – All rows end with True.
  – Example: $B \rightarrow A \lor B$
Determining formula type

• How long does it take to check if a formula is satisfiable?
  – If somebody gives you a satisfying assignment, then in time roughly the size of the formula.
  • On a m-symbol formula, take time $O(m) = \text{constant} \ast m$, for some constant depending on the computer/software.

– What if you don’t know a satisfying assignment? How hard it is to find it?
  • Using a truth table: in time $O(m \ast 2^n)$ on a length m n-variable formula.
  • Is it efficient?...
Complexity of computation

• Would you still consider a problem really solvable if it takes very long time?
  – Say $10^n$ steps on an n-symbol string?
  – At a billion ($10^9$) steps per second (~1GHz)?
  – To process a string of length 100...
  – will take $10^{100}/10^9$ seconds, or $\sim 3 \times 10^72$ centuries.

  – Age of the universe: about $1.38 \times 10^{10}$ years.
  – Atoms in the observable universe: $10^{78}$-$10^{82}$. 
Complexity of computation

• What strings do we work with in real life?
  – A DNA string has $3.2 \times 10^9$ base pairs
  – A secure key in crypto: 128-256 bits
  – Number of Walmart transactions per day: $10^6$.
  – URLs searched by Google in 2012: $3 \times 10^{12}$. 
Determining formula type

• How long does it take to check if a formula is satisfiable?
  – Using a truth table: in time $O(m \times 2^n)$ on a length $m$ n-variable formula.
  – Is it efficient?
    • Not really!
    • Formula with 100 variables is already too big!
    • In software verification: millions of variables!

– Can we do better?

A million-dollar question!
Formula simplification

• Equivalent formulas:
  – Have the same truth table.

• If two formulas $F$ and $G$ are equivalent, then can substitute $F$ for $G$ (and vice versa) in any formula $H$.
  
  \[ A \land C \rightarrow (\neg B \lor C) \]
  
  \[ \text{We know: } (p \rightarrow q) \text{ is equivalent to } (\neg p \lor q) \]

  \[ A \land C \rightarrow (\neg B \lor C) \text{ is equivalent to: } \neg(A \land C) \lor (\neg B \lor C) \]

• But now it looks inconvenient, with that negation on the outside... Can we make it simpler?
Let A be “it’s sunny” and B “it’s cold”.

• “It’s sunny and cold today”! -- No, it’s not!

• That could mean
  – No, it’s not sunny.
  – No, it’s not cold.
  – No, it’s neither sunny nor cold.

• In all of these scenarios, “It’s either not sunny or not cold” is true.
The law of excluded middle

• In classical logic, the law of excluded middle say that either a statement or its opposite must be true.
• But here by the opposite we really mean a negation

- A: It is sunny.
- ¬A: It is not sunny
- A: Today is Tuesday.
- ¬A: Today is not Tuesday
- A: John votes for NDP.
- ¬A: John does not vote for NDP
- A: You are with us
- ¬A: You are not with us.
De Morgan’s Laws

• What is the negation of a longer logic statement?
  – Take a truth table column and flip all the values.

• Some useful simplifications: De Morgan’s laws.
  – For AND: $\neg (A \land B)$ is equivalent to $(\neg A \lor \neg B)$
  – For OR: $\neg (A \lor B)$ is equivalent to $(\neg A \land \neg B)$

• Example:
  – $\neg (\neg A \lor B)$ is $\neg\neg A \land \neg B$, same as $A \land \neg B$
  – So, since $(A \to B)$ is equivalent to $(\neg A \lor B)$,
    $\neg (A \to B)$ is equivalent to $A \land \neg B$
Knights and knaves

- On a mystical island, there are two kinds of people: knights and knaves. Knights always tell the truth. Knaves always lie.

- Puzzle 2: You see three islanders talking to each other, Arnold, Bob and Charlie. You ask Arnold “Are you a knight?”, but can’t hear what he answered. Bob pitches in: “Arnold said that he is a knave!” and Charlie interjects “Don’t believe Bob, he’s lying”. Out of Bob and Charlie, who is a knight a who is a knave?