cse541
LOGIC FOR COMPUTER SCIENCE

Professor Anita Wasilewska

Spring 2015

LECTURE 1

GENERAL INFORMATION

Course Web Page
www.cs.stonybrook.edu/~ cse541

The webpage contains

Course Syllabus

Book Chapters

Lectures slides

Sample Exercises and Problems with solutions

Sample Tests

Course Text Book

AN INTRODUCTION TO CLASSICAL
and NON-CLASSICAL LOGICS

Anita Wasilewska

Book Chapters and Lecture Slides are in Downloads on the course web page

This is a book in writing; all correction, remarks by students are welcome!

Additional Books:

Introduction to Mathematical Logic, Fourth Edition, Elliot Mendelson, Wadsworth&Brooks/Cole Advanced Books &Software, Pacific Grove, California

A Friendly Introduction to Mathematical Logic, C.C. Leary, Prientice Hall, 2000

Course Goal

The goal of the course is to make student understand the need of, and to learn the formality of logic as scientific field

The book is written with students on my mind so that they can read and learn by themselves, even before coming to class

The goal the course and of the book is to teach not only understanding of classical and some non- classical logics but to teach also what is a formal logic as scientific subject, with its languages, definitions, problems and basic theorems

Tests

There will be TWO MIDTERMS and a FINAL examination

I will also give TWO PRACTICE TESTS for extra credit

Midterms 1, 2 (100pts each)

Final (100pts)

Practice Midterm 1 (10 extra points) - given in class

Practice Final (10 extra points) - take home

Course Webpage contains many examples and exercises from the text book and solutions for posted Exercises-Homework Problems for you to **study** from

I also posted some previous TESTS (with solutions) so you could see the **format** of tests

GRADES for the tests will depend on the form, details, and carefulness of written solutions.

Grading

During the semester you can earn **300pts** or more (in the case of extra points). The grade will be determine in the following way:

# of earned points divided by 3 = % grade

The % grade is translated into letter grade in a standard way i.e.

100 - 90 % is A range;

A (100-96%), A- (95- 90%)

89 - 80 % is B range:

B- (80 - 83%), B (84 -86%), B+ (87 -89%)

79 - 70 % is C range:

C- (70- 72%), C (73-76%), C+(77-79%)

69 - 60 % is D range

F is below 60%

None of grades will be curved

Course Contents and Schedule

The course will follow the book very closely and in particular we will cover some, or all of the following subjects chapters

Part one

Motivation, history, syntax and semantics for classical propositional calculus.

Formal symbolic propositional languages, formal definitions of **model, counter model, tautology** for **propositional logic**

Part two

Some Many Valued Extensional Semantics

Course Contents and Schedule

Part three

Formal deductive systems, called also **proof systems**

General definition and examples. Definition of a formal proof.

Relationship between proof systems and their semantics, i.e general definition of notions of **soundness and completeness** of a given proof systems relatively to given semantics.

Definition of a logic as a complete proof system

Part four

Hilbert style proof systems for classical propositional logic. Proofs of DEDUCTION theorem, and two different proofs of the COMPLETENESS theorem for propositional classical logic.

Course Contents and Schedule

Part five

Automated Gentzen type proof systems 1:

RS proof system for classical propositional logic

Examples of the automatic proof-search

Automated Gentzen type proof systems 2:

Original Gentzen proof system

Part six

A Hilbert style proof system for Intuitionistic Logic

Relationship between Intuitionstic and Classical logics

Course Contents and Schedule

Part seven

Automated proof systems 3:

Gentzen proof system for Intuitionistic Logic. Heuristic decision procedures.

Part eight

Languages and semantics for classical **predicate logic**

Hilbert Proof systems and proof of completeness theorem

Course Contents and Schedule

Part nine

Automated Gentzen type proof systems 4:

QRS proof system for classical predicate logic.

Examples of the automatic proof-search.

Constructive proof of COMPLETENESS theorem

Original Gentzen proof system for classical and Intuitionistic predicate Logics.

Part ten

A Hilbert style proof systems for Modal Logics S4 and S5

Relationships with Intuitionistic Logic.

Chapter 1
INTRODUCTION

PART 1: Logic for Mathematics:

Logical Paradoxes

PART 2: Logic for Mathematics:

Semantical Paradoxes

PART 3: Non- Classical Logics and

Logics for Computer Science

PART 4: Computer Science Puzzles

Chapter 1
PART1: Mathematical Paradoxes

Early Intuitive Approach:

Until recently, till the end of the 19th century, mathematical theories used to be built in the intuitive, or axiomatic way.

Historical development of mathematics has shown that it is not sufficient to base theories only on an intuitive understanding of their notions

Example

Consider the following.

By a set, we mean intuitively, any collection of objects.

For example, the set of all even integers or the set of all students in a class.

The objects that make up a set are called its members (elements)

Sets may themselves be members of sets for example, the set of all sets of integers has sets as its members

Example

Sets may themselves be members of sets for example, the set of all sets of integers has sets as its members

Most sets are not members of themselves;

the set of all students, for example, is not a member of itself, because the set of all students is not a student

However, there may be sets that do belong to themselves - for example, the set of all sets

Russell Paradox

Consider the set A of all those sets X such that X is not a member of X

Clearly, A is a member of A if and only if A is not a member of A

So, if A is a member of A, the A is also not a member of A; and if A is not a member of A, then A is a member of A

In any case, A is a member of A and A is not a member of A.

CONTRADICTION!

Russell Paradox Solution

Russel proposed his Theory of Types as a solution to the Paradox

The idea is that every object must have a definite non-negative integer as its type assigned to it

An expression x is a member of the set y is meaningful if and only if the type of y is one greater than the type of x

Russell Paradox Solution

Russell's theory of types guarantees that it is meaningless to say that a set belongs to itself.

Hence Russell' s solution is:

The set A as stated in the Russell paradox does not exist

The Type Theory was extensively developed by by Whitehead and Russell in years 1910 - 1913

It is successful, but difficult in practice and has certain other drawbacks as well

LOGICAL PARADOXES

Logical Paradoxes, also called Logical Antinomies are paradoxes concerning the notion of a set

A a modern development of Axiomatic Set Theory as one of the most important fields of modern Mathematics , or more specifically Mathematical Logic , or Foundations of Mathematics resulted from the search for solutions to various Logical Paradoxes

First paradoxes free axiomatic set theory was developed by Zermello in 1908

LOGICAL PARADOXES

Two of the most known antinomies, other then Russell 's Paradox are Cantor and Burali-Forti antinomies

They were stated at the end of 19th century

Cantor Paradox involves the theory of cardinal numbers

Burali-Forti Paradox is the analogue to Cantor's but in the theory of ordinal numbers

Cardinality of Sets

We say that sets X and Y have the same cardinality, $cardX = cardY$ or that they are equinumerous if and only if there is one-to-one correspondence that maps X onto Y

$cardX \leq cardY$ means that X is equinumerous with a subset of Y

$cardX < cardY$ means that $cardX \leq cardY$ and $cardX \neq cardY$

Cantor and Schröder- Berstein Theorems

Cantor Theorem

For any set X,
$cardX < card\mathcal{P}(X)$

Schröder- Berstein Theorem

For any sets X and Y,

If $cardX \leq cardY$ and $cardY \leq cardX$, then $cardX = cardY$.

Ordinal numbers are the numbers assigned to sets in a similar way as cardinal numbers but they deal with ordered sets

Cantor Paradox, 1899

Let C be the universal set - that is, the set of all sets

Now, $\mathcal{P}(C)$ is a subset of C, so it follows easily that

$$card\mathcal{P}(C) \leq cardC$$

On the other hand, by Cantor Theorem,

$$cardC < card\mathcal{P}(C) \leq card\mathcal{P}(C)$$

so also

$$cardC \leq card\mathcal{P}(C).$$

From Schröder- Berstein theorem we have that $card\mathcal{P}(C) = cardC$, what contradicts Cantor Theorem

Solution: **Universal set does not exist.**

Burali-Forti Paradox, 1897

Given any ordinal number, there is a still larger ordinal number.

But the ordinal number determined by the set of all ordinal numbers is the largest ordinal number

Solution: **the set of all ordinal numbers do not exist**

Logical Paradoxes

Another solution to Logical Paradoxes:

Reject the assumption that for every property P(x), there exists a corresponding set of all objects x that satisfy $P(x)$

Russell's Paradox then simply proves that **there is no set** A defined by a property $P(x)$: of all sets that do not belong to themselves

Logical Paradoxes

Cantor Paradox shows that

there is no set A defined by a property

$P(X)$: there is an universal set X

Burali-Forti Paradox shows that

there is no set A defined by a property

$P(x)$: there is a set that contains all ordinal numbers

Intuitionism

A more **radical interpretation** of the paradoxes has been advocated by Brower and his **intuitionist school**

Intuitionists refuse to accept the universality of certain basic logical laws, such as the law of **excluded middle: A or not A**

For intuitionists the **excluded middle law** is true for finite sets, but it is invalid to extend it to all sets.

The intuitionists' concept of infinite set differs from that of classical mathematicians

Intuitionists' Mathematics

The basic difference between **classical** and **intuitionists' mathematics** lies in the interpretation of the word **exists**

In classical mathematics proving **existence** of an object x such that $P(x)$ holds **does not mean** that one is able to indicate a method of **construction** of it

In the **intuitionists' universe** we are justified in asserting the **existence** of an object having a certain property only if we know an **effective method** for constructing, or finding such an object

Intuitionists' Mathematics

In intuitionist' mathematics the paradoxes are **not derivable**, or even meaningful

The Intuitionism, because of its **constructive** flavor, has found a lot of applications in computer science

For example in the theory of programs correctness.

Intuitionistic Logic (to be studied in this course) reflects intuitionists ideas in a form a formalized deductive system

PART 2: SEMANTIC PARADOXES

SEMANTIC PARADOXES

The development of axiomatic theories solved some,

but not all problems brought up by the Logical Paradoxes.

Even the consistent sets of axioms, as the following examples show, do not prevent the occurrence of another kind

of paradoxes, called Semantic Paradoxes that deal with the notion of truth.

SEMANTIC PARADOXES

Berry Paradox, 1906:

Let A denote the set of all positive integers which can be defined in the English language by means of a sentence containing at most 1000 letters

The set A is finite since the set of all sentences containing at most 1000 letters is finite. Hence, there exist positive integer which do not belong to A.

Consider a sentence: *n* is the least positive integer which cannot be defined by means of a sentence of the English language containing at most 1000 letters

This sentence contains **less than 1000 letters** and defines a positive integer *n*

Therefore $n \in A$ - but $n \notin A$ by the definition of *n*

CONTRADICTION!

Berry Paradox Analysis

The paradox resulted entirely from the fact that **we did not say** precisely what **notions and sentences** belong to the arithmetic and what **notions and sentences** concern the arithmetic

Of course we didn't talk about and examine arithmetic as a fix and closed deductive system

We also **incorrectly mixed** the natural language with mathematical language of arithmetic

Berry Paradox Solution

We have to distinguish always the **language of the theory** (arithmetic) and the language which **talks about the theory**, called a metalanguage

In general **we must distinguish a theory from the meta-theory**

In well and correctly defined theory the such paradoxes can not appear

The Liar Paradox

A man says: I am lying.

If he is lying , then what he says is true, and so he is not lying

If he is not lying, then what he says is not true,

and so he is lying

CONTRADICTION!

Liar Paradoxes

These paradoxes arise because the concepts of the type

" I am true", " this sentence is true", " I am lying"

should not occur in the **language** of the theory

They belong to a **metalanguage** of the theory

It it means they belong to a language that talks **about the theory**

Cretan Paradox

The Liar Paradox is a corrected version of a following paradox stated in antiquity by a Cretan philosopher **Epimenides**

Cretan Paradox

The Cretan philosopher Epimenides said: All Cretans are liars

If what he said is **true** , then, since Epimenides is a Cretan, it must be **false**

Hence, what he said is false. Thus, there is a Cretan who is not a liar

CONTRADICTION with what he said: ”All Cretans are liars”

GENERAL REMARKS;
The Goal of the Course

FIRST TASK when one builds mathematical logic foundations of mathematics or of computer science is to define formally and proper **symbolic language**

This is called building a proper syntax

SECOND TASK is to extend the syntax to include a **notion of a proof**

It allows us to find out what can and cannot be proved if certain axioms and rules of inference are assumed

This part of syntax is called **PROOF THEORY**

GENERAL REMARKS;
The Goal of the Course

THIRD TASK is to define formally what does it mean that formulas of our formal language defined in the TASK ONE **are true**

It means that we have to define what we formally call a **semantics** for our **language**

For example, the notion of truth i.e. the **semantics** for the **classical** and **intuitionistic** approaches are different

GENERAL REMARKS;
The Goal of the Course

FOUTH TASK is to investigate the relationship between **proof theory** (part of the syntax) and **semantics** for the given language

It means to establish correct relationship between notion of a **proof** and the notion of **truth**, i.e. to answer the following questions

Q1: Is (and when) everything one proves is true?

Q2: Is it possible (and when it is possible) to guarantee provability of everything we know to be true ?

GENERAL REMARKS;
The Goal of the Course

The GOAL of this course is to formally define and develop the above Four Tasks in case of the **Classical Logic** and in case of **Non- Classical Logics** like Intuitionistic Logic, some Modal Logics, and some Many Valued Logics

Chapter 1
PART 3: Non- Classical Logics and Logics for Computer Science

Logics in Computer Science

The use of Classical Logic in **computer science** is known, indisputable, and well established.

The existence of PROLOG and Logic Programming as a separate field of computer science is the best example of it.

Intuitionistic Logic in the form of Martin-Löf's theory of types (1982), provides a complete theory of the process of program specification, construction, and verification.

A similar theme has been developed by Constable (1971) and Beeson (1983)

Modal Logics

In 1918, an American philosopher, C.I. Lewis proposed yet another interpretation of lasting consequences, of the logical implication.

In an attempt to avoid, what some felt, the paradoxes of implication (a false sentence implies any sentence) he created a modal logic.

The idea was to distinguish **two sorts of truth**: necessary truth and mere possible (contingent) truth

A possibly true sentence is one which, though true, could be false

Modal Logics for Computer Science

Modal Logics in Computer Science are used as as a tool for analyzing such notions as knowledge, belief, tense.

Modal logics have also been also employed in a form of Dynamic logic (Harel 1979) to facilitate the statement and proof of properties of programs

Non-classical Logics for Computer Science

Temporal Logics were created for the specification and verification of concurrent programs (Harel, Parikh, 1979, 1983),

for a specification of hardware circuits (Halpern, Manna, Maszkowski, (1983)),

and also to specify and clarify the concept of causation and its role in commonsense reasoning Shoham, 1988

Fuzzy Sets, Rough Sets, Many valued logics were created and developed to reasoning with incomplete information.

Non-classical Logics for Computer Science

The development of new logics and the applications of logics to different areas of Computer Science and in particular to Artificial Intelligence is a subject of a course in itself but is beyond the scope of class.

In class we will examine in detail the Classical Logic and some aspects of the Intuitionistic Logic

We also introduce some of the most standard many valued, and some modal logics

Chapter 1

PART 4: Computer Science Puzzles

Computer Science Puzzles
Reasoning Artificial Intelligence

Reasoning in Artificial Intelligence

Assumption 1:

Flexibility of reasoning is one of the key property of intelligence

Assumption 2:

Commonsense inference is **defeasible** in its nature;

Assumption 3:

we are all capable of drawing conclusions, acting on them, and then **retracting** them if necessary in the face of new evidence

Reasoning in Artificial Intelligence

If Computer programs are to act intelligently, they will need to be similarly **flexible**

Goal:

development of formal systems (logics) that describe commonsense flexibility.

Flexible Reasoning Example

Reiter, 1987

Consider a statement Birds fly. Tweety, we are told, is a bird.

From this, and the fact that birds fly, we conclude that Tweety can fly

This conclusion is **defeasible:** Tweety may be an ostrich, a penguin, a bird with a broken wing, or a bird whose feet have been set in concrete.

Non-monotonic Inference:

on learning a new fact (that Tweety has a broken wing), we are forced to **retract** our conclusion (that he could fly)

Non-monotonic Logics

Definition:

A **non-monotonic** logic (reasoning) is a logic in which the introduction of a new information (axioms) can **invalidate** old facts (theorems)

Definition:

A **default** reasoning (logic) is a reasoning that let us draw of plausible inferences from less-then- conclusive evidence in the absence of information to the contrary

Observe: non-monotonic reasoning is an example of the default reasoning

Non-monotonic Logics

Here is what is happening in our CS department NOW!

Tiantian Gao will give his RPE presentation on

Controlled Natural Languages and Default Reasoning

this Thursday Aug 28 at 1:30pm in Room 1310

Part of the **Abstract**

Controlled natural languages (CNLs) are effective languages for knowledge representation and reasoning.

Over the past 20 years, a number of machine-oriented CNLs emerged and have been used in many application domains for **problem solving and question answering**

However, few of them support **nonmonotonic inference**

In our work, we propose **nonmonotonic extensions of CNL** to support **defeasible reasoning**

Believe, Auto-epistemic, Logics

Example: Moore, 1983

Consider my reason for believing that I do not have an older brother.

It is surely not that one of my parents once casually remarked, You know, you don’t have any older brothers, nor have I pieced it together by carefully sifting other evidence.

I simply believe that if I did have an older brother I would know about it;

therefore since I don’t know of any older brothers of mine, I must not have any

Auto-epistemic, Logics

The brother example reasoning is not default reasoning nor non-monotonic reasoning

It is a reasoning about one's own knowledge or **belief**

Definition

Any reasoning about one's own knowledge or **belief** is called an **auto-epistemic** reasoning

Auto-epistemic reasoning **models** the reasoning of an ideally rational agent reflecting upon his beliefs or knowledge

Logics which describe it are called auto-epistemic logics

Computer Science Puzzle
Missionaries and Cannibals Revisited

McCarthy, 1985

Here is the old Cannibals Problem:

Three missionaries and three cannibals come to the river.

A rowboat that seats two is available.

If the cannibals ever outnumber the missionaries on either bank of the river, the missionaries will be eaten.

How shall they cross the river?

Traditionally the puzzler is expected to devise a strategy of rowing the boat back and forth that gets them all across and avoids the disaster.

Traditional Solution

A state is a triple comprising the number of missionaries, cannibals and boats on the starting bank of the river.

The initial state is 331 , the desired state is 000

A solution is given by the sequence:

331, 220, 321, 300, 311, 110, 221, 020, 031, 010, 021, 000.

Missionaries and Cannibals Revisited

Imagine now giving someone a problem, and after **he puzzles** for a while, he suggests going upstream half a mile and crossing on a bridge.

What a bridge? you say.

No bridge is mentioned in the statement of the problem.

He replies: Well, they don't say the isn't a bridge.

So you modify the problem to exclude the bridges and pose it again.

He proposes a helicopter, and after you exclude that, he proposes a winged horse....

Missionaries and Cannibals Revisited

Finally, you tell him the solution.

He attacks your solution on the grounds that the boat might have a leak.

After you rectify that omission from the statement of the problem, he suggests that a see monster may swim up the river and may swallow the boat

Finally, you must look for a mode of reasoning that will settle his hash once and for all.

McCarthy Solution

McCarthy proposes circumscription as a technique for solving his puzzle.

He argues that it is a part of **common knowledge** that a boat can be used to cross the river **unless** there is something with it or something else **prevents** using it

If our facts do not require that there be something that prevents crossing the river, the **circumscription** will generate the conjecture that there isn't

Lifschits has shown in 1987 that in some special cases the **circumscription** is equivalent to a first order sentence.

In those cases we can go back to our secure and well known classical logic.