AL-HIKMAH UNIVERSITY, ILORIN, NIGERIA
Adeta Road, Adewole Housing Estate, P.M.B. 1601, Ilorin
…..learning for wisdom and morality…..
CENTRE FOR ICT AND DISTANCE LEARNING (IDL)
e-CONTENT DEVELOPMENT (DL) UNIT

e-Note for SEN204

A. Course Lecturer's Details
Name: OYELAKIN, A. Moruff
Mobile Number:+2348062738520
Official Email Address:amoyelakin@alhikmah.edu.ng

B. Faculty Department and Programme
Faculty: Natural and Applied Sciences
Department: Computer Science
Programme: B.Sc. Computer Science
C. Course Title: Logic and Its Applications in Computer Science
D. Course Credit: 03

E. Course Description

Valid & invalid arguments; translating from English to the language of propositional and predicate logic; formal deduction and its role in the validity of an argument; logic & computer science – how to build circuits from logic gates and how to minimize circuits using propositional logic; introduction to prolog – a programming language based on logic; and, the applications of logic in computer science- A1, automated theorem-provers, expert systems.

F. Learning Objectives

At the completion of this e-note, it s expected that learners will achieve the following:

1. Learn some basic concepts on the Logic (Deductive reasoning, valid arguments, invalid arguments and so on)
2. Learn and apply some concepts in logic and their applications in Computer Science
3. Master some real life arguments and their representation in logical forms
4. Understand propositional statements, connectives and propositional logic in different forms
5. Identify the importance and application areas of Logic in Computer Science

G. Notes on the course

This course is one of the mathematical-based courses in Computer Science. As a software engineering student, you are required to have a mastery of how to encode and interpret information. The nature of this course requires that students have a good understanding of some basic concepts in Logic. Then, one will be able to follow the points explained herein.

Deductive Reasoning

Deduction is a method of reasoning from the general to the specific. This is also called deductive reasoning and top-down logic. In a deductive argument, a conclusion follows necessarily from the stated premises. (Contrast with induction). In logic, a deductive argument is called a syllogism. In rhetoric, the equivalent of the syllogism is the enthymeme.

Examples

"The fundamental property of a deductively valid argument is this: If all of its premises are true, then its conclusion must be true also because the claim asserted by its conclusion already has been stated in its premises, although usually only implicitly.

Validity and Invalidity, Soundness and Unsoundness

The task of an argument is to provide statements (premises) that give evidence for the conclusion. There are two basic kinds of arguments. They are deductive argument and Inductive argument.

Deductive argument: This involves the claim that the truth of its premises guarantees the truth of its conclusion; the terms valid and invalid are used to characterize deductive arguments. A deductive argument succeeds when, if you accept the evidence as true (the premises), you must accept the conclusion.

Inductive argument: This involves the claim that the truth of its premises provides some grounds for its conclusion or makes the conclusion more probable; the terms valid and invalid cannot be applied.

Valid: an argument is said to be valid if and only if it is necessary that if all of the premises are true, then the conclusion is true; if all the premises are true, then the conclusion must be true; it is impossible that all the premises are true and the conclusion is false.

A deductive argument is said to be valid if and only if it takes a form that makes it impossible for the premises to be true and the conclusion nevertheless to be false. Otherwise, a deductive argument is said to be invalid.

Invalid: an argument that is not valid. We can test for invalidity by assuming that all the premises are true and seeing whether it is still possible for the conclusion to be false. If this is possible, the argument is invalid.

Validity and invalidity apply only to arguments, not statements. For our purposes, it is just nonsense to call a statement valid or invalid. True and false apply only to statements, not arguments. For our purposes, it is just nonsense to call an argument true or false. All deductive arguments aspire to validity. If you consider the definitions of validity and invalidity carefully, you'll note that valid arguments have the following important property: valid arguments preserve truth. If all your premises are true and you make a valid argument from them, it must be the case that whatever conclusion you obtain is true. (We shall see below, however, that valid arguments do not necessarily preserve truth value: it is entirely possible to argue validly from false premises to a true conclusion).

Sound: an argument is sound if and only if it is valid and contains only true premises. A deductive argument is sound if and only if it is both valid, and all of its premises are actually true. Otherwise, a deductive argument is unsound

Unsound: an argument that is not sound.

Counterexample: an example which contradicts some statement or argument (ex. a counterexample to the statement "All fifteen year-olds have blue hair" would be a fifteen-year-old without blue hair); for an argument, a counterexample would be a situation in which the premises of the argument are true and the conclusion is false; counterexamples show statements to be false and arguments to be invalid.

Hints on establishing the validity and strength of arguments

When evaluating arguments, we have two main questions to ask:

1. Do the premises provide enough logical support for the conclusion?
2. Are the premises true?

In this segment, we'll discuss how to answer the first question. As we mentioned, the answer to this question depends on whether the argument is deductive or non-deductive.

Validity applies to deductive arguments, strength applies to non-deductive arguments.

More notes on deductive arguments and validity

- A **valid** argument is a deductive argument that succeeds in providing decisive logical support.

A valid argument is thus a deductive argument – an argument that attempts to establish conclusive support for its conclusion – that succeeds.

- An **invalid** argument is a deductive argument that fails in providing conclusive support.
- Definition: A **strong** argument is a non-deductive argument that succeeds in providing probable, but not conclusive, logical support for its conclusion. A **weak** argument is a non-deductive argument that fails to provide probable support for its conclusion.

If an argument is weak, you'd be better off throwing a coin to know if the conclusion is true, and that's far from succeeding in providing reasons for a conclusion. So if the conclusion is unlikely to be true when the premises are true, then the argument is weak. Game over.

But how are we to decide when an argument is strong and not weak?

The answer to this question is contextual. As a lecturer, my standards are very strict. I'm extremely pedantic, and I'm going to point out every mistake you made. My goal is to make sure that you learn from your mistakes. But if I do that when I'm at a party with my friends, I would not be very popular, now would I? I need to change my standards there. When I'm in the court of law, I need to have strong standards, beyond reasonable doubt. So establishing that an argument is strong in court is quite demanding. We want to minimise the mistakes we make.

A Brief Explanation on Strong or weak arguments

97% of vegetarians are healthy. Madison is a vegetarian. Therefore, Madison is probably healthy.

If the premises are true, what are the chances that Madison is healthy? 97%! She might not be. She might be amongst the 3% that are not healthy, but it's quite unlikely. So if the premises are true, it's very likely that the conclusion is also true—but it may be false. Hence, this argument is strong.

Bailey's father is a plumber, so Bailey's father has a van.

I think this one is also strong. Here's my reasoning. Being a plumber is the kind of job that requires you to go around with tools, and tools that are messy and get dirty and everything, and you don't want to put it in a nice little car; you need space. In my experience, plumbers typically have vans. So then it would be quite reasonable to expect Bailey's father to have a van if he's a plumber.

Some Examples on Validity and Soundness (of an argument)

According to the definition of a deductive argument (see the Deduction and Induction), the author of a deductive argument always intends that the premises provide the sort of justification for the conclusion whereby if the premises are true, the conclusion is guaranteed to be true as well. Loosely speaking, if the author's process of reasoning is a good one, if the premises actually do provide this sort of justification for the conclusion, then the argument is valid.

In effect, an argument is valid if the truth of the premises logically guarantees the truth of the conclusion. The following argument is valid, because it is impossible for the premises to be true and the conclusion nevertheless to be false:

Elizabeth owns either a Honda or a Saturn.
Elizabeth does not own a Honda.
Therefore, Elizabeth owns a Saturn.

It is important to stress that the premises of an argument do not have actually to be true in order for the argument to be valid. An argument is valid if the premises and conclusion are related to each other in the right way so that if the premises were true, then the conclusion would have to be true as well. We can recognize in the above case that even if one of the premises is actually false, that if they had been true the conclusion would have been true as well. Consider, then an argument such as the following:

All toasters are items made of gold.
All items made of gold are time-travel devices.
Therefore, all toasters are time-travel devices.

Obviously, the premises in this argument are not true. It may be hard to imagine these premises being true, but it is not hard to see that if they were true, their truth would logically guarantee the conclusion's truth.

It is easy to see that the previous example is not an example of a completely good argument. A valid argument may still have a false conclusion. When we construct our arguments, we must aim to construct one that is not only valid, but sound. A sound argument is one that is not only valid, but begins with premises that are actually true. The example given about toasters is valid, but not sound. However, the following argument is both valid and sound:

In some states, no felons are eligible voters, that is, eligible to vote.
In those states, some professional athletes are felons.
Therefore, in some states, some professional athletes are not eligible voters.

Here, not only do the premises provide the right sort of support for the conclusion, but the premises are actually true. Therefore, so is the conclusion. Although it is not part of the definition of a sound argument, because sound arguments both start out with true premises and have a form that guarantees that the conclusion must be true if the premises are, sound arguments always end with true conclusions.

It should be noted that both invalid, as well as valid but unsound, arguments can nevertheless have true conclusions. One cannot reject the conclusion of an argument simply by discovering a given argument for that conclusion to be flawed.

Whether or not the premises of an argument are true depends on their specific content. However, according to the dominant understanding among logicians, the validity or invalidity of an argument is determined entirely by its logical form. The logical form of an argument is that which remains of it when one abstracts away from the specific content of the premises and the conclusion, that is, words naming things, their properties and relations, leaving only those elements that are common to discourse and reasoning about any subject matter, that is, words such as "all," "and," "not," "some," and so forth. One can represent the logical form of an argument by replacing the specific content words with letters used as place-holders or variables.

For example, consider these two arguments:

All tigers are mammals.
No mammals are creatures with scales.
Therefore, no tigers are creatures with scales.

All spider monkeys are elephants.
No elephants are animals.
Therefore, no spider monkeys are animals.

These arguments share the same form:

All A are B;
No B are C;
Therefore, No A are C.

All arguments with this form are valid. Because they have this form, the examples above are valid. However, the first example is sound while the second is unsound, because its premises are false. Now consider:

All basketballs are round.
The Earth is round.
Therefore, the Earth is a basketball.

All popes reside at the Vatican.
John Paul II resides at the Vatican.
Therefore, John Paul II is a pope.

These arguments also have the same form:

All A's are F;
X is F;
Therefore, X is an A.

Arguments with this form are invalid. This is easy to see with the first example. The second example may seem like a good argument because the premises and the conclusion are all true, but note that the conclusion's truth isn't guaranteed by the premises' truth. It could have been possible for the premises to be true and the conclusion false. This argument is invalid, and all invalid arguments are unsound.

While it is accepted by most contemporary logicians that logical validity and invalidity is determined entirely by form, there is some dissent.

The logical form of a statement is not always as easy to discern as one might expect. For example, statements that seem to have the same surface grammar can nevertheless differ in logical form. Take for example the two statements:

(1) Tony is a ferocious tiger.
(2) Clinton is a lame duck.

Despite their apparent similarity, only (1) has the form "x is a A that is F." From it one can validly infer that Tony is a tiger. One cannot validly infer from (2) that Clinton is a duck. Indeed, one and the same sentence can be used in different ways in different contexts. Consider the statement:

(3) The King and Queen are visiting dignitaries.

It is not clear what the logical form of this statement is. Either there are dignitaries that the King and Queen are visiting, in which case the sentence (3) has the same logical form as “The King and Queen are playing violins,” or the King and Queen are themselves the dignitaries who are visiting from somewhere else, in which case the sentence has the same logical form as “The King and Queen are sniveling cowards.” Depending on which logical form the statement has, inferences may be valid or invalid. Consider:

The King and Queen are visiting dignitaries. Visiting dignitaries is always boring. Therefore, the King and Queen are doing something boring.

Only if the statement is given the first reading can this argument be considered to be valid.

Because of the difficulty in identifying the logical form of an argument, and the potential deviation of logical form from grammatical form in ordinary language, contemporary logicians typically make use of artificial logical languages in which logical form and grammatical form coincide. In these artificial languages, certain symbols, similar to those used in mathematics, are used to represent those elements of form analogous to ordinary English words such as “all”, “not”, “or”, “and”, and so forth. The use of an artificially constructed language makes it easier to specify a set of rules that determine whether or not a given argument is valid or invalid. Hence, the study of which deductive argument forms are valid and which are invalid is often called “formal logic” or “symbolic logic.”

In short, a deductive argument must be evaluated in two ways. First, one must ask if the premises provide support for the conclusion by examine the form of the argument. If they do, then the argument is valid. Then, one must ask whether the premises are true or false in actuality. Only if an argument passes both these tests is it sound. However, if an argument does not pass these tests, its conclusion may still be true, despite that no support for its truth is given by the argument.

Note: there are other, related, uses of these words that are found within more advanced mathematical logic. In that context, a formula (on its own) written in a logical language is said to be valid if it comes out as true (or “satisfied”) under all admissible or standard assignments of meaning to that formula within the intended semantics for the logical language. Moreover, an axiomatic logical calculus (in its entirety) is said to be sound if and only if all theorems derivable from the axioms of the logical calculus are semantically valid in the sense just described.

For a more sophisticated look at the nature of logical validity, see the articles on “Logical Consequence” in this encyclopedia. The articles on “Argument” and “Deductive and Inductive Arguments” in this encyclopedia may also be helpful.

The validity and strength of arguments

When evaluating arguments, we have two main questions to ask:

1. Do the premises provide enough logical support for the conclusion?
2. Are the premises true?

In this segment, we'll discuss how to answer the first question. As we mentioned, the answer to this question depends on whether the argument is deductive or non-deductive.

Validity applies to deductive arguments, strength applies to non-deductive arguments.

Let's start with deductive arguments and validity.

- Definition: A **valid** argument is a deductive argument that succeeds in providing decisive logical support.

A valid argument is thus a deductive argument – an argument that attempts to establish conclusive support for its conclusion – that succeeds.

- Definition: An **invalid** argument is a deductive argument that fails in providing conclusive support.

For deductive arguments, you answer "yes" to the question "Do the premises provide enough logical support for the conclusion?" if the argument is valid, and you answer "no" if otherwise.

Take the following deductive argument:

Patrick's jeans are blue, therefore, Patrick's jeans are coloured.

Is it possible for the premises to be true and the conclusion to be false? If my jeans are blue, then they have a colour. If they have a colour, then they're coloured. Hence, it is impossible for the premise (Patrick's jeans are blue) to be true, and the conclusion (Patrick's jeans are coloured) to be false. Therefore, the argument is valid.

Another Interest Illustration on Arguments

If you throw a dice, either it lands on six or it doesn't. So the dice has a 50% chance of landing on six.

Some people believe that, but this is an invalid argument. What is the probability for a dice to land on six? There are six faces and the dice is likely to land on any of them. Since six only shows on one face, there's only a one out of six chance that the dice will land on six. And one out of six is a lot less than 50%. It is thus possible for the premise of the argument to be true, but the conclusion false.

Arguments can be valid even if they are rubbish:

If there is a purple elephant in the hall, then I am a giant turkey. There is a purple elephant in the hall. Therefore, I'm a giant turkey.

This argument is nonsensical, but it's valid. If the premises were true, the conclusion would be guaranteed to be true. You need to be careful here. 'Valid' does not necessarily mean good or bad. It just means succeeding in establishing conclusive support for its conclusion. Of course, the premises of this argument are false. But claiming that an argument is valid is not to claim that the premises are true. Validity is about succeeding in providing conclusive support for the conclusion, if the premises were true.

For non-deductive arguments, we don't talk about valid and invalid arguments, we talk instead about strong and weak arguments.

- Definition: A **strong** argument is a non-deductive argument that succeeds in providing probable, but not conclusive, logical support for its conclusion.
- A **weak** argument is a non-deductive argument that fails to provide probable support for its conclusion.

If an argument is weak, you'd be better off throwing a coin to know if the conclusion is true, and that's far from succeeding in providing reasons for a conclusion. So if the conclusion is unlikely to be true when the premises are true, then the argument is weak. Game over.

But how are we to decide when an argument is strong and not weak?

The answer to this question is contextual. As a lecturer, my standards are very strict. I'm extremely pedantic, and I'm going to point out every mistake you made. My goal is to make sure that you learn from your mistakes. But if I do that when I'm at a party with my friends, I would not be very popular, now would I? I need to change my standards there. When I'm in the court of law, I need to have strong standards, beyond reasonable doubt. So establishing that an argument is strong in court is quite demanding. We want to minimise the mistakes we make.

In a deductive argument, validity is the principle that if all the premises are true, the conclusion must also be true. Also known as formal validity and valid argument.

In logic, validity isn't the same as truth. As Paul Tomassi observes, "Validity is a property of arguments. Truth is a property of individual sentences. Moreover, not every valid argument is a sound argument"Valid arguments” are valid by virtue of their form" (although not all logicians would wholly agree). Arguments that are not valid are said to be invalid.

Some clarifications and explanations on arguments

"A **formally valid argument** that has true premises is said to be a sound argument. In debate or discussion, therefore, an argument may be attacked in two ways: by attempting to show that one of its premises is false or by attempting to show that it is invalid. On the other hand, if one concedes the truth of the premises of a formally valid argument, one must also concede the truth of the conclusion--or be guilty of irrationality."

- "I once heard former RIBA President Jack Pringle defend flat roofs with the following syllogism: We all like Edwardian terraces. Edwardian terraces use curtain walls to hide their sloping roofs and pretend they're flat. Ergo: we must all like flat roofs.
  "Except that we don't, and they still leak."

- **Analyzing the Validity of an Argument**
  "The primary tool in deductive reasoning is the syllogism, a three-part argument consisting of two premises and a conclusion.

Examples

All Rembrandt paintings are great works of art.
The Night Watch is a Rembrandt painting.
Therefore, The Night Watch is a great work of art.
All doctors are quacks.
Smith is a doctor.
Therefore, Smith is a quack.

"The syllogism is a tool for analyzing the **validity** of an argument. You'll rarely find a formal syllogism outside of textbooks on logic.

Four Valid Argument Forms

"There are a great many **valid argument** forms, but we shall consider only four basic ones. They are basic in the sense that they occur in everyday use, and that all other valid argument forms can be derived from these four forms:

1. Affirming the Antecedent
   If p then q
   p Therefore, q
2. Denying the Consequent
   If p then q
   Not-q Therefore, not-p
3. Chain Argument
   If p then q
   If q then r Therefore, if p then r
4. Disjunctive Syllogism
   Either p or q
   Not-p Therefore, q

Note:

Whenever we find an argument whose form is identical to one of these valid argument forms, we know that it must be a valid argument.

Logic/ Propositional Logic-A Quick Introduction

Logic is the basis of all mathematical reasoning, and of all automated reasoning. The rules of logic specify the meaning of mathematical statements. These rules help us understand and reason with statements.

Importance of Logic in Computing -Computer Science as a course is deeply rooted in Mathematics. The rules of logic give precise meaning to mathematical statements. These rules are used to distinguish between valid and invalid mathematical arguments. Apart from its importance in understanding mathematical reasoning, logic has numerous applications in Computer Science, varying from design of digital circuits, to the construction of computer programs and verification of correctness of programs.

What is a proposition?

A proposition is the basic building block of logic. It is defined as a declarative sentence that is either True or False, but not both.

The **Truth Value** of a proposition is True(denoted as T) if it is a true statement, and False(denoted as F) if it is a false statement. For Example,

1. The sun rises in the East and sets in the West.
2. 1 + 1 = 2
3. 'b' is a vowel.

All of the above sentences are propositions, where the first two are Valid(True) and the third one is Invalid(False).
Some sentences that do not have a truth value or may have more than one truth value are not propositions.
For Example,

1. What time is it?
2. Go out and play.
3. x + 1 = 2.

The above sentences are not propositions as the first two do not have a truth value, and the third one may be true or false.

To represent propositions, **propositional variables** are used. By Convention, these variables are represented by small alphabets such as .
The area of logic which deals with propositions is called **propositional calculus** or **propositional logic**.
It also includes producing new propositions using existing ones. Propositions constructed using one or more propositions are called **compound propositions**. The propositions are combined together using **Logical Connectives** or **Logical Operators**.

Truth Table in detail

In Logic, we have what we call Truth Table. A Truth Table shows the possible combination of some set of input values to produce a set of output values. These values are generally known as truth values. Since we need to know the truth value of a proposition in all possible scenarios, we consider all the possible combinations of the propositions which are joined together by Logical Connectives to form the given compound proposition. This compilation of all possible scenarios in a tabular format is called a truth table**.**

To be able to draw a valid truth table, some basic concepts should be understood. For instance, the general formula for knowing the number of values to draw in a truth table is 2^n, where n is the number of input values. If there are two input values are two, we use 2^2 (2 raise to power 2).
Lets assume the input values are X and Y, then the truth table can be of the form:

X	Y
T	T

T	F
F	T
F	F

The other concept that learner must understand fully to be able to draw valid truth table is propositional connective or connective.

More notes on Propositional or Logical Connectives

A propositional connective is used to establish the relationship that exits among two or more arguments or statement. Our focus in this course will be on three propositional connectives. These are: Disjunction, Conjunction and Negation. We have some other ones such as Exclusive OR, Implication and so on.

Connective	**Symbol**	**Interpretation**
Disjunction	V	OR
Conjuction	$\wedge$	AND
Negation	$\neg$	NOT

So, if we want to draw a Truth table that shows the relationship among two input variables X1 and X2 using disjunction connective (that is OR gate).
The solution is as shown below:

X1	**X2**	**X1 V X2**
T	T	T
T	T	T
T	F	T
T	F	T
F	T	T
F	T	T
F	F	F
F	F	F

Another example is when we have two input variables A and B joined together using conjunction connective.
The solution is as shown below.

X1	X2	X 1 $\wedge$ X2
T	T	T
T	F	F

F	T	F
F	F	F

For the negation connective, the focus is to achieve the opposite of a statement. If the statement is in the negative, it changes to positive. Similarly, if the statement is positive, it changes to negative.

Supposing we have a variable P, P’ or P or ¯ P ¬

P	P’ or ¬ P
T	F
F	T

As a way to understand truth table, logic and connectives (connectors) better, let us have a table summarizing how each are used in Logic.

Name of the connector	Symbol	Connection	Meaning
AND (Conjunction)	∧	P ∧ Q	(P∧Q) is true if both P **and** Q are true
OR (Disjunction)	∨	P ∨ Q	(P∨Q) is true if **either** of them is true
Negation	¬	¬P	¬P is the **opposite** of P
Exclusive OR	⊕	P ⊕ Q	(P⊕Q) is true if **either** of them is true but **both are not** at the same time
Implication	→	P → Q	(P→Q) implies **if** P is true **then** Q is true.
Double Implication	↔	P ↔ Q	(P↔Q) implies P will be true **if and only if** Q is true.

Application areas of Logic in Artificial Intelligence, Theorem Proofing and Expert Systems

Logic has been found to be very useful in several areas of Artificial Intelligence, Theorem Proofing and Expert System Applications.

Experts have identified three uses of logic in Artificial Intelligence. These applications include: as a tool of analysis, as a basis for knowledge representation, and as a programming language. A large part of the effort of developing limited-objective reasoning systems goes into the management of large, complex bodies of declarative information.

Artificial Intelligence is the subfield of Computer Science devoted to developing programs that enable computers to display behavior that can (broadly) be characterized as intelligent. Throughout its relatively short history, AI has been heavily influenced by logical ideas. AI has drawn on many research methodologies: the value and relative importance of logical formalisms is questioned by some leading practitioners, and has been debated in the literature from time to time.

Computational Logic is the process of designing and analyzing logic in computer applications. In It is important for students to learn how to create logic based on the statements and constraints provided. Logic in relation to computers is mainly of two types: Propositional Logic and First Order Logic (FOL). Our emphasis at this level is on propositional logic. This has been described to some extent in previous sections.

Logic has equally been widely used for the following areas of Computer Science:

i. Formal verification
ii. theorem proving
iii. proof assistants,
iv. and for computable functions

Knowledge Representation

In response to the need to design this declarative component, a subfield of AI known as knowledge representation emerged during the 1980s. Knowledge representation deals primarily with the representational and reasoning challenges of this separate component.

Logic has equally been found to be very useful in Knowledge representation from the time immemorial.

Practical-based demonstration on Prolog Programming Environments

Prolog is a logic programming language associated with artificial intelligence and computational linguistics. The language has its roots in first-order logic, a formal logic, and unlike many other programming languages, Prolog is intended primarily as a declarative programming language: the program logic is expressed in terms of relations, represented as facts and rules. A computation is initiated by running a query over these relations.

The language was developed and implemented in Marseille, France in 1972 by Alain Colmerauer with Philippe Roussel, based on Robert Kowalski's procedural interpretation of Horn clauses

A version of Prolog that has been very popular is SWI-Prolog. SWI-Prolog has become a popular Free Software implementation of the Prolog language. IT is recommended that we practice introductory codes in prolog using this tool. The tool has the graphical development environment as well as the command line tools which we see as vital to the success of the system.

Another version of Prolog tool that can be used to get started is **Visual Prolog**. This tool was formerly known as **PDC Prolog** and **Turbo Prolog**. It is a strongly typed object-oriented extension of Prolog. As Turbo Prolog, it was marketed by Borland but it is now developed and marketed by the Danish firm Prolog Development Center (PDC) that originally developed it. Visual Prolog can build Microsoft Windows GUI-applications, console applications, DLLs (dynamic link libraries), and CGI-programs.

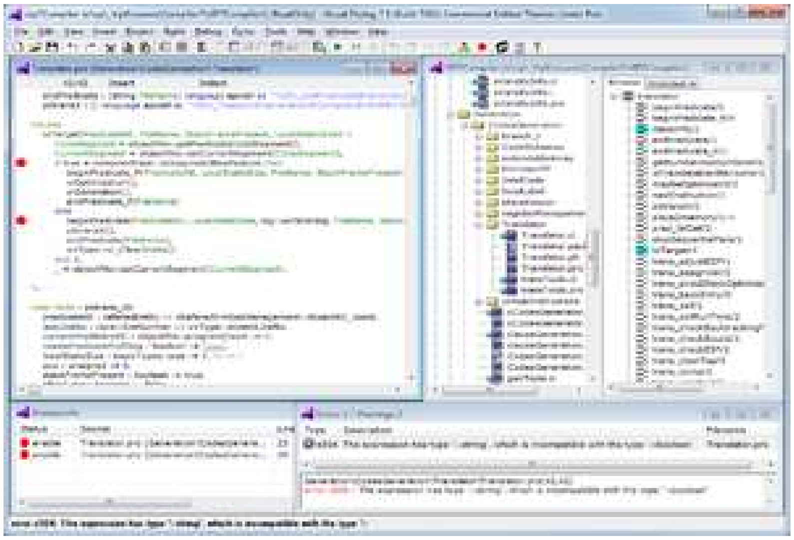

Figure 1: Visual Prolog/PDC Prolog Environment

As found in the literature, some of the application areas of Prolog in building intelligent applications in Computer Science are:

i. intelligent data base retrieval
ii. natural language understanding
iii. expert systems
iv. specification language
v. machine learning
vi. robot planning
vii. automated reasoning
viii. problem solving

Steps for Programming in Prolog

i. declare facts describing explicit relationships between objects and properties objects might have (e.g. Mary likes pizza, grass has_colour green, Fido is_a_dog, Mizuki taught Paul Japanese)
ii. define rules defining implicit relationships between objects (e.g. the sister rule above) and/or rules defining implicit object properties (e.g. X is a parent if there is a Y such that Y is a child of X).
iii. One then uses the system by: asking questions above relationships between objects, and/or about object properties (e.g. does Mary like pizza? is Joe a parent?)

Handling Relations in Prolog

Prolog programs specify *relationships* among objects and properties of objects. When we say, "John owns the book", we are declaring the ownership relationship between two objects: John and the book. When we ask, "Does John own the book?" we are trying to find out about a relationship.

Relationships can also rules such as:

a. Two people are sisters **if**
b. they are both female **and**
   they have the same parents.

A rule allows us to find out about a relationship even if the relationship isn't explicitly stated as a fact.

Note: Efforts will be made during lectures to demonstrate how to use a free source Prolog tool to write elementary programs.

H. Study Questions (For practice only)

1. Define each of the following:

(i) Argument
(ii) Argument validity
(iii) Soundness of argument
(iv) Propositional connective
(v) Propositional logic

2. List and explain any four propositional connectives that are used in logic

3. Identify any four areas of applications of Logic in Computer Science

4. How does critical thinking help a Computer Scientist in the discharge of his duties?

5. Given three variables P1, P2, and P3. You are required to design a truth table that shows the relationship among the variables using one disjunction connective and conjunction connective.

6. Write short note on formally valid argument

7. Itemise any seven examples of propositional statement in real life situation

8. Write some statements to demonstrate argument validity and soundness

9. List and discuss three areas of application of Logic in Artificial Intelligence

10. Briefly differentiate between valid argument and sound argument, giving relevant examples.

I. References

1. Aiello, Luigia Carlucci, Doyle, Jon, and Shapiro, Stuart (eds.), 1996, KR'96: Principles of Knowledge Representation and Reasoning, San Francisco: Morgan Kaufmann.
2. Clocksin, William F.; Mellish, Christopher S. (2003). Programming in Prolog. Berlin ; New York: Springer-Verlag. ISBN 978-3-540-00678-7.
3. Bratko, Ivan (2012). Prolog programming for artificial intelligence (4th ed.). Harlow, England ; New York: Addison Wesley. ISBN 978-0-321-41746-6.
4. Stanford Encyclopedia of Philosophy (2018). Logic and Artificial Intelligence, retrieved from https://plato.stanford.edu/entries/logic-ai/
5. Sarah Skwire and David Skwire (2014).Writing With a Thesis: A Rhetoric and Reader, 12th ed. Wadsworth, Cengage,
6. William Hughes and Jonathan Lavery (2004). Critical Thinking: An Introduction to the Basic Skills. Broadview Press

J. Recommended books/Materials /Web Documents

1. William Hughes and Jonathan Lavery (2004). Critical Thinking: An Introduction to the Basic Skills. Broadview Press, 2004
2. Aiello, Luigia Carlucci, Doyle, Jon, and Shapiro, Stuart (eds.), 1996, KR'96: Principles of Knowledge Representation and Reasoning, San Francisco: Morgan Kaufmann.
3. Clocksin, William F.; Mellish, Christopher S. (2003). Programming in Prolog. Berlin ; New York: Springer-Verlag. ISBN 978-3-540-00678-7.
4. Bratko, Ivan (2012). Prolog programming for artificial intelligence (4th ed.). Harlow, England ; New York: Addison Wesley. ISBN 978-0-321-41746-6.