International Journal of Engineering, Science and Mathematics
Vol. 6 Issue 3, July 2017,
ISSN: 2320-0294 Impact Factor: 6.765
Journal Homepage: http://www.ijesm.co.in, Email: ijesmj@gmail.com Double-Blind
Peer Reviewed Refereed Open Access International Journal - Included in the International Serial Directories Indexed & Listed at: Ulrich's Periodicals Directory ©, U.S.A., Open J-Gage as well as in Cabell's Directories of Publishing Opportunities, U.S.A

MAJOR PROBLEM – SOLVING APPLICATIONS OF DISCRTE MATHEMATICS USING LOGIC OPERATIONS IN COMPUTER SCIENCE: A CRITICAL STUDY

Sidhu[1], Dr. Amardeep Singh[2]
Department of Mathematics
[1,2]Shri Venkateshwara University, Gajraula (Uttar Pradesh)

Abstract

Discrete mathematics is the section of science focused to the study of discrete objects. In this research, we critically analyze "problem solving applications of discrete math's with the help of logical operation in computer. These axioms are made through the laws of discrete science. It has the applications to all fields of computer science; it is used broadly in telecommunications and information processing. In discrete mathematics, we are concerned with objects such as integers, propositions, sets, relations and functions which are all discrete. We determine and study the concepts associated with them, properties and relationships among them. Discrete mathematics includes sets, functions and relations, matrix algebra, combinatory and finite probability, graph theory, finite differences and recurrence relations, logic, mathematical induction, and algorithmic thinking. It is the mathematical language of data science, and as such, its importance has heightened dramatically in recent decades.The applications of the discrete mathematics state the context of set theory, algebra and calculus.Discrete math forms the substructure of many real world scientific fields especially computer science. The primary techniques learned in a discrete math course can be applied to many different fields.

1. OVERVIEW

Discrete mathematics for software engineering", it studies that discrete science in legitimate activity of software engineering. It presents incomplete requests, well-established sets and complete enlistment. The standard of acceptance applies to sets with a requesting undeniably progressively complex that the requesting on the characteristic numbers. As far it express that an application, an application, it demonstrate the interesting prime factorization in Z and talk about GCD's ideas of combinatorics as far as checking issues. It likewise presents the binomial and multinomial coefficients and investigations a portion of their properties. It finishes up with the Inclusion-Exclusion Principle. The idea confirmation by acceptance and its capacity characterize the capacity of recursion. However, PC projects may not end for all info, so the idea of halfway capacity is pivotal. Henceforth, we characterize cautiously relations, capacities and fractional capacities and examine a portion of their properties (being injective, emotional, bijective).[1].

What is computational thinking" investigate computational speculation to return to the essential logical inquiries of registering. It is a critical way to deal with takes care of issues, planning or making frameworks and understanding human conduct that draws on ideas of key to registering. Computational reasoning is a sort of logical reasoning. It imparts to scientific speculation in the general manners by which we may approach taking care of an issue. It imparts to building thinking in the general manners by which we may approach structuring and assessing an enormous, complex framework that works inside the limitations of this present reality. It imparts to logical deduction in the general manners by which we may approach getting calculability, insight, the brain and human conduct. It additionally show and present inquiries in registering as an approach to advise ourselves that there are logical difficulties that underlie our individual research interests or advancements in innovation.[2]

2. Discrete Mathematics with application

It investigates the significant topics of discrete mathematics, yet in addition the thinking that underlies numerical idea. Understudies build up the capacity to think conceptually as they study the thoughts of rationale and confirmation. While finding out about such ideas as rationale circuits and PC expansion, calculation examination, recursive reasoning, process ability, automata, cryptography, and combinatorics, understudies find that the thoughts of discrete mathematics underlie and are basic to the science and innovation of the PC age. By and large, Epp's accentuation on thinking furnishes understudies with a solid establishment for software engineering and upper-level science courses. The ALU is the center of the PC - it performs math and rationale activities on information that not just understand the objectives of different applications (e.g., scientific logical and designing projects), yet additionally control addresses (e.g., pointer arithmetic). In this segment, we will review calculations utilized for the fundamental mathematics and logical operations. A key supposition that will be that twos supplement portrayal will be utilized, except if generally noted. Boolean Addition: which exceeds the modulus of 2 for Boolean numbers ($\mathbf{B} = \mathbf{Z}_2 = \{0,1\}$, the integers modulo 2). Thus, we record a zero for the sum and propagate a carry valued at one into the next more significant digit.

It provides a clear in concepts such as logic circuits, computer addition, algorithm analysis, recursive thinking and combinatories. Discrete mathematics are essential to the science and technology age. It is important to note that Epp's emphasis on reasoning provides a strong foundation in computer science. [3]

3. Context of discrete mathematics in circuit design

The idea and systems of discrete arithmetic in circuit graph. This takes a gander at another utilization of formal reasoning and a mathematical way to deal with software engineering: the utilization of discrete mathematics to help with the way toward planning advanced circuits.

Equipment configuration isn't the genuine subject here, so we will clarify only enough about equipment to clarify the specific circuits engaged with the exchange. Notwithstanding applying discrete science to determine and simulate circuits.

The combination of discrete mathematics makes it conceivable to do a few valuable errands: exact specification of circuits, simulation, correctness proofs, and circuit inductions. Advanced circuit configuration is a huge branch of knowledge, and there isn't space here to cover every last bit of it. Along these lines we will think about just one class of computerized circuits (combinational circuits, which don't contain signals). Discrete mathematics is utilized intensely all through the whole subjects of computerized circuit plan and PC design. We apply the techniques for Propositional Logic to circuit configuration, incorporating prevailing upon truth tables and logarithmic thinking about circuits.[4]

The graph theory

- Researchers principally and mainly use graphs to model problems as diverse as how to detect a deadlock condition in an operating system or working framework. It is likewise helps, how to design productive routings for transportation systems. Some pivotal issues in man-made brainpower utilize productive looking through strategies on the class of charts called trees as viable technique.. This introduces graphs as a structure that is used to represent or show and solve many problems in a variety of areas in computer science.
- The graph theory mainly organized into four main parts. The first introduces the terms that are used to describe this structure.
- The second spotlights on ideas, thoughts dealing with graphs that are connected. As an application of this idea, we will examine and analyze Euler's theorem, the primary theorem of graph theory. This theorem clarifies when it is conceivable, for example, to pass across a series of bridges and return to the starting point without crossing any bridge twice. The theorem can advise how to plot a chart without taking the pen off the paper too often times.
- The other part covers the uncommon diagrams graphs called trees. Trees are a typical information structure in software engineering that is utilized to take care of looking and arranging issue. Finally, the last part of the chapter deals with directed graphs, which extend the idea of a graph to include the notion of direction. Directed graphs are used to represent and solve problems such as scheduling a production of subassemblies with no unnecessary delays or designing a one-way street grid. [5]

Functions in Programming

The 'function as a graph' concept used to model a function mathematically is not actually equivalent as a function written in a programming language. Albeit, both are relations of the equivalent concept.. The diff erence is that a set of ordered pairs specifies only what result

should be produced for each input; there is no concept of an algorithm that can be used to obtain the result. The function graph approach in a programming language is represented solely by the algorithm and understanding, and the only way to determine the value of (fx) is to execute the algorithm on input x. A programming language function is a method for computing results; a mathematical function is a set of answers. . Other than giving a technique to getting the outcome, a programming language capacity has conduct:: it consumes memory and time in order to compute the result of an application. For example, we might write two sorting functions, one that takes very little time to run on a given test sample and one that takes a long time.

We would regard them as diff erent algorithms and calculations, and would concentrate on that diff erence as being important. The graph model of a function lacks any notion of speed and would make no distinction between the two algorithms as long as they always produce the same results. There are several important classes of functions defined by algorithms, which we will examine in the next few sections. The essential questions we are interested in are the termination and execution speed of the function.

Ourresearch questions and describes the methodology aimed at developing a "functional definition" of discrete mathematics. Our research is inscribed in a "contemporary epistemology".

"The fundamental idea is that discrete mathematics is the study of mathematical structures that are "discrete" in contrast with "continuous" ones. The discrete structures are the setups, which can be portrayed with a limited or countable arrangement of relations.Albeit, discrete mathematics acquires particular objects and methods of problem solving.Discrete arithmetic gets specific items and strategies for critical thinking .However, these attempts to define discrete mathematics are not all-inclusive as they overlook many characteristics of the concepts and proofs involved in this field. In further assessment, what is additionally significant for the didactics of science is to reveal the specificities of this field of mathematics in contrast with others In further assessment, what is additionally significant for the didactics of science is to reveal the specificities of this field of arithmetic in contrast with others

Discrete mathematics continues to be promoted and advanced as the essential mathematics in a 21st century. Its power lies in the opportunity it provides for supporting reasoning, problem solving, modelling, and systematic thinking in the field of research and computing.. Besides, recursion and recursive thinking seem to be powerful modelling and problem solving strategies throughout mathematics in general in learning of discrete mathematics in particular. The last has been featuredin the studies of recursion and recursive thinking. They depict, describe the integration of recursive thinking with iterative as well as algebraic thinking, and they present the benefits of this integration as means to deepen the students understanding of each of the geometry of transformations and co-variation of variables.[6]

5. CONCLUSION

In understanding discrete mathematics we should comprehend the results presented here, as it came to the conclusion that the DM material should be understood with examples and applications from computer science, because the applications would enhance the understanding of discrete mathematics.The major aim of logic in computer science is to develop languages to model the situations we encounter as computer science professionals, in such a condition that we can reason about them formally. Reasoning about situations means constructing arguments about them; we want to do this formally, so that the arguments are valid and can be defended rigorously, or executed on a machine.

In understanding mathematics we must comprehend, what makes a correct mathematical argument, a proof is a succession of steps which prompts to the desired conclusion Proofs are used to verify that computer programs produce the correct result, to establish the security of a system and to create artificial intelligence. Logic is interested in true or false statements and how the truth or falsehood of a statement can be determined from other statements

REFERENCES

[1]. Jean Galler (2008) Discrete Mathematics For Computer Science, Research Gate Publication
[2]. J. M. Wing (2006) Computational Thinking, Communications Of The ACM
[3]. S. Epp. Susanna (2004) Discrete Mathematics With Applications. Thomson-Brooks/ Cole
[4]. John O'Donnell, Cordelia Hall and Rex Page (2006**)** DicsreteMthematics Using A Computer,Springer Publication
[5]. BiswaPurna Chandra (2005) Discrete Mthematics And Graph Theory, Prentice Hall Of Private Limited
[6]. Ari Ben Mordechai, "Mathematical logic for computer science, Springer