Journal of Information Systems Engineering and Management
2025, 10(24s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Exam Scheduling using Graph Coloring

Dr Meghana N. Vyas[1], Dr G B Hemalatha[2]
[1]Assistant Professor, Department of Mathematics & Statistics, Somaiya Vidyavihar University
meghna.vyas@somaiya.edu
ORCID ID: 0009-0006-5465-1177
[2]Assistant Professor, Department of CS & IT, Somaiya Vidyavihar University
ORCID ID: 0009-0004-2902-314X
hemalatha@somaiya.edu

ARTICLE INFO

Received: 25 Dec 2024

Revised: 13 Feb 2025

Accepted: 27 Feb 2025

ABSTRACT

Throughout the academic year, educational institutions use an official and structured document called an academic calendar to communicate and arrange important dates and events. It functions as a thorough timetable that delineates the exact dates of terms, semesters, and sessions that are dedicated to educational activities. The dates of the semesters and terms, deadlines for registration, class schedules, exam periods, holidays and breaks, dates of graduation, special events and academic activities, administrative deadlines etc. are all commonly included in the academic calendar. A crucial component of educational institutions is the examination procedure, which is scheduled with dates and hours decided upon before the academic year starts.

An educational institution's course schedule and exam schedule, which require a combination of resources including teachers, subjects, students, and classrooms, are the most significant components of the academic calendar. The academic institute creates the course calendar and exam schedule with the fewest possible resources in order to avoid conflicts by meeting different priority and essential requirements. Graph coloring issues, which are NP complete problems, can be used to solve this scheduling challenge. In this study, under certain limitations, Exam scheduling was created using graph coloring.

Keywords: Graph theory, Graph coloring, Academic calender, Examination Schedule, conflicts.

INTRODUCTION:

The Koinsber bridge problem from 1735 served as the catalyst for the development of graph theory. The Eulerian Graph notion originated from this difficulty. Euler created a structure known as the Eulerian graph in order to solve the Koinsberg bridge problem. A.F. Mobius proposed the concepts of complete and bipartite graphs in 1840, while Kuratowski used recreational problems to demonstrate that they are planar. Gustav Kirchhoff developed the idea of a tree, or a connected graph without cycles, in 1845. He used concepts from graph theory to determine the currents in electrical networks and circuits. Thomas Gutherie discovered the renowned four color problem just once in 1852.

Later, in 1856, William R. Hamilton and Thomas P. Kirkman examined cycles on polyhydra and created the idea of a Hamiltonian graph by examining journeys that made exactly one stop at a given location. H. Dudeney brought up a puzzle issue in 1913. Even though the four color dilemma was created, Kenneth Appel and Wolfgang Haken weren't able to solve it until a century later. This period is regarded as the inception of graph theory. When Sylvester first coined the term "Graph" in 1878, he compared "Quantic invariants" to covariants of algebra and molecular diagrams. In mathematics, graph theory is the study of graphs, which are mathematical structures used to model pairwise relations between objects.

DEFINITIONS:

A Graph: A pictorial representation of a set of objects where some pairs of objects are connected by links, usually denoted G(V,E) or G = (V,E) – consists of set of vertices V together with a set of edges E.

Vertices: The interconnected objects are represented by points termed as vertices.

Edges: The links that connect the vertices are called edges.

Adjacent vertices: Two vertices u and v are adjacent if there exists an edge (u,v) that connects them.

Simple Graph: If the graph G having only one edge between any two vertices, then the graph is called simple graph.

Multiple Graph: If the graph G having more than one edge between any two vertices, then the graph is called Multiple graphs.

Directed Graph: A graph in which the edges have a direction.

Undirected Graph: A graph in which the edges does not have a direction.

Indegree of vertex: The total count of incoming edges to a particular vertex is known as the indegree of that vertex.

Outdegree of vertex: The total number of outgoing edges from a particular vertex is known as the outdegree of that vertex.

Graph coloring: A graph coloring is an assignment of labels, called colors, to the vertices of a graph such that no two adjacent vertices share the same color.

Numerous tasks can be accomplished with the graph coloring, including scheduling sports, creating seating arrangements, scheduling exams, creating schedules or timetables, assigning mobile radio frequencies, resolving Sudoku puzzles, allocating registers, coloring maps, arranging taxis, and more.

Chromatic number: The chromatic number $\chi(G)$ of a graph G is the minimal number of colors for which such an assignment is possible.

Flowchart: A diagram that represents a workflow or process. A flowchart can also be defined as a diagrammatic representation of an algorithm, a step-by-step approach to solving a task.

REVIEW OF LITERATURE

With the goal of creating an exam schedule using graph coloring under specific constraints, Angshu Kumar Sinha et al. (2022) studied the fundamentals of graph theory and the graph coloring technique. They then developed an algorithm for creating an exam schedule that has fewer exam days without any conflicts. They have further verified this algorithm using a variety of software programs, including Python, C, and C++.

In order to better understand how graph theory is applied in everyday life, S. Venu Madhava Samara (2012) first explained all the basic concepts of graph theory, including graphs, vertex, edges, walks, cycles, loops, adjacency graphs, adjacency matrices, subgraphs, connected graphs, graph sequences, trees, and spanning trees, using examples. This was followed by a brief explanation of Euler circuits, Euler graphs, Hamiltonian circuits, and Hamiltonian graphs. Additionally, the author used graph theoretical planning approaches to build a GSM (Group Special Mobile) for the Bharat Sanchar Nigam Limited. These techniques were used for scheduling timetables, solving traveling salesman problems, coloring maps, and vertex coloring.

Piotr Formaowicz and Kraysztof Tanas (2012), The authors have presented a survey of graph coloring as an important subfield of graph theory, taking into account that graph coloring is one of the most well-known, well-liked, and thoroughly researched subjects with many applications and conjectures that are still being explored and studied by mathematicians and computer scientists worldwide. They do this by describing various methods of coloring, such as vertex coloring, equitable vertex coloring, circular vertex coloring, acyclic vertex colouring, star vertex coloring, edge coloring, circular edge coloring, Acyclic edge coloring, berge Fulkerson and fan raspaud coloring, face and ma coloring, list coloring, path coloring, and total coloring. Ultimately, they developed an algorithmic method for coloring graphs.

Arifin, Indra Bayu Muktyas, et al.'s work provides a thorough examination of the theoretical underpinnings, previous research, and pertinent studies that support the development of the Graph Coloring Algorithm for exam scheduling. This study discusses several approaches, such as modifying pre-existing methods, applying a graph coloring technique improved by bitwise operations, and developing an effective exam scheduling framework. Furthermore, the study discusses useful applications in educational settings, bitwise operations' ability to improve speed, scheduling alternatives visualization, and lays the groundwork for further studies on graph colouring methods.

OBJECTIVES

1. To Recognize the importance of Graph theory.

2. To understand the fundamentals of Graph Coloring.

3. To develop design to perform graph coloring.

4. To use the concepts of Graph Coloring to construct exam schedule.

RESEARCH METHODOLOGY:

The methodology revolves around developing an exam schedule that complied with specific requirements, particularly for students with a greater number of elective subjects who aimed to complete their exams in the least amount of time possible. This initiative was intended to optimize their time management, enabling them to advance to subsequent studies. The challenge was approached using principles from graph theory, which led to the creation of an algorithm that devised an exam schedule. This schedule ensured the minimum number of days was utilized while covering all elective courses without any subject clashes.

THE REAL PROBLEM

Graph coloring was applied to create exam schedules for M.Sc. Mathematics students who have chosen different course combinations. In order to accomplish this, we first describe the problem as a graph in which the vertices represent the courses, and the edge connecting two vertices indicates whether or not a common student exists for the courses they represent. Suppose the color corresponds to a "day of examination." Thus, arranging the test is equivalent to coloring the conflict graphs with the fewest possible colors. If two students are studying the same subject, there is a dispute.

The non-conflicting graph is an additional one, in which the edges are drawn in a course that is mutually exclusive and does not match any pupil. It is occasionally observed that non-conflicting graphs from specified input sets and constraints facilitate time savings. While some situations only require a few resources, others could call for several at once. As was previously indicated, a schedule might contribute to several parts of the issue. When educators work with resources, additional matters like topic content and instructor availability are addressed. To generate a comprehensive schedule, further data input in the form of each teacher's chosen area must be supplied.

NOTATIONS

Let $G = (V, E)$ be a simple connected graph with vertex set V and edge set E. Consider the function $f : V \rightarrow \{\text{color } 1, \text{color } 2, \text{color } 3, \ldots, \}$ such that the two vertices vi and vj are adjacent if $f(v_i) \neq f(v_j)$ or if $f(v_i) = f(v_j)$ then the vertices vi and vj are nonadjacent vertices. The minimum number of color required to coloring the graph is called chromatic number of the graph and is denoted by $\chi(G)$.

ALGORITHM

The suggested algorithm's key idea is to use the minimum amount of colors so that every two neighboring vertices have a distinct color to calculate the conflict graphs' chromatic numbers. test routines at schools, colleges, and universities are created by algorithms that operate based on the subjects that students take courses in common but have distinct test dates. In this case, the minimal color usage corresponds to the number of exam dates needed to carry out the exam.

A formal description of the algorithm is given below.

Exam time Scheduling Problem using Graph Coloring (ETSPGC)

Input: Conflict graph G (V, E)

Output: Chromatic number of graph $\chi(G)$

Step 1: Determine the graph G's degree sequence.

Step 2: Select the vertex of maximum degree vi If the maximum degree vertex having more than one then select one of them, vi (say)

Step 3: Assign a color to the maximum degree vertex vi

Step 4: Assign same colors to the vertex non-adjacent to the vertices vi .

Step 5: Verify if any two nearby vertices have the same color. If so, apply a different color to the non-adjacent vertex of the vi that hasn't been used yet; if not, move on to the next step.

Step 6: Select next maximum degree vertex vj (say) which is not yet colored

Step 7: Assign different color which is not yet used to the vertex vj .

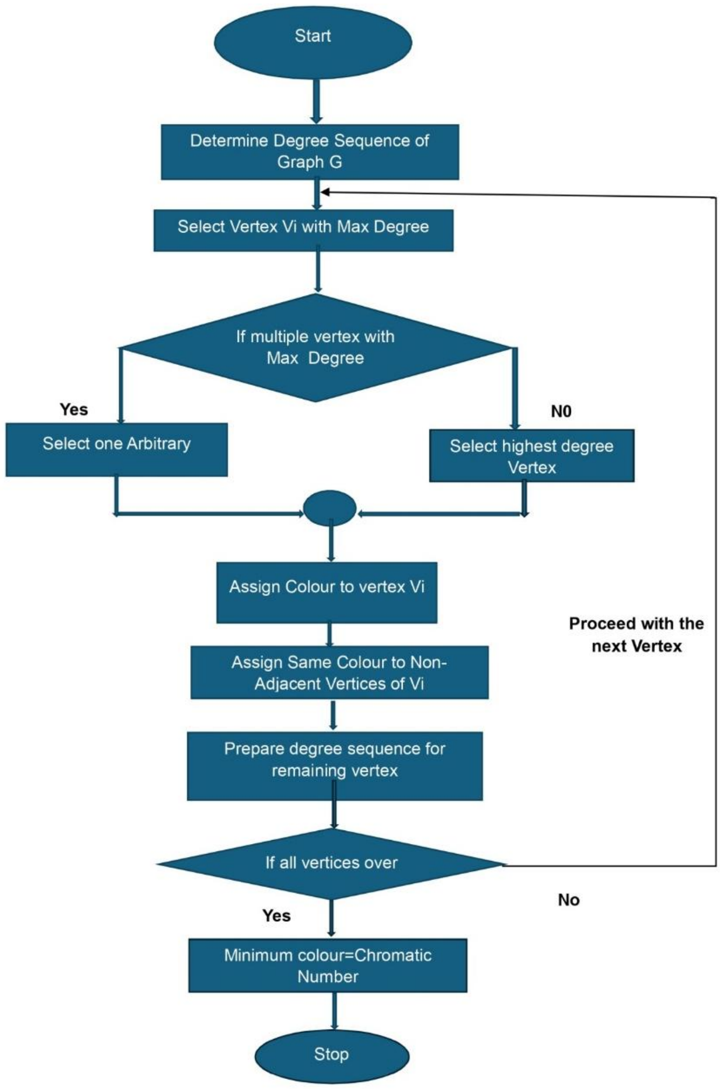
Step 8: If any two non-adjacent vertices vl and vk have different color If all the adjacent vertices of vl and vk are assigned different color then assign same color to the vertex vl and vk . Else go to step 9

Step 9: Repeat step 2 to step 5 until all vertices are colored.

Step 10: Compute the minimum number of colors use.

Step 11: Chromatic number (G).

Flowchart: the flowchart for finding the chromatic number of above algorithm :



AN ALGORITHMIC ILLUSTRATION

Students pursuing an M.Sc. in Mathematics during a specific semester are assigned the following combination of subjects.

Table 1: Subject Combinations

Subject 1	**Subject 2**	**Subject 3**	**Subject 4**	**Subject 5**
Advanced complex analysis	Differential Geometry	Operations Research	Special functions	
Advanced complex analysis	Differential Geometry	Algebra	Ordinary differential equations	Ring and module theory
Operations Research	Functional analysis	Partial differential equations	Algebra	
Ring and module theory	Partial differential equations	Ordinary differential equations	Special functions	

Our goal in solving this problem is to determine the bare minimum of exam days needed to schedule exams for the nine subjects so that there are no conflicts for students taking any of the specified subject combinations. Additionally, come up with a plan that utilizes a minimum of days.

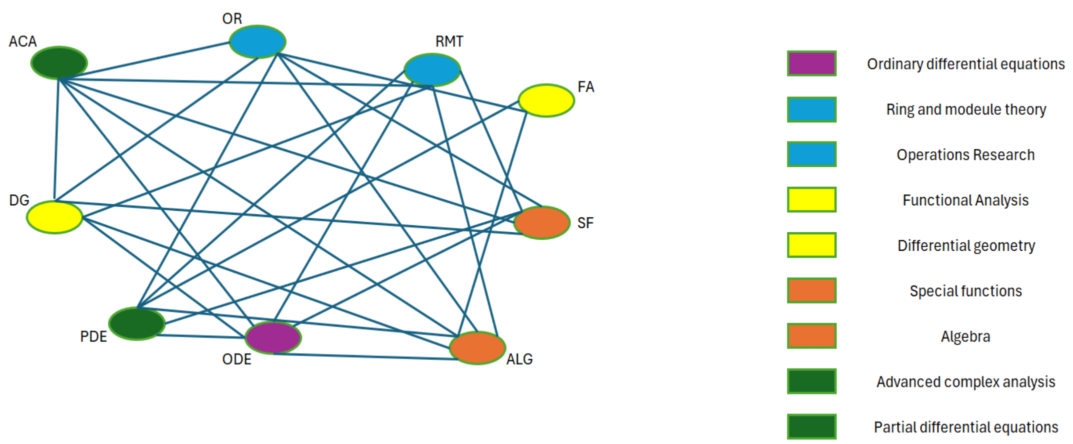


Figure : 1 : Graph Coloring

Draw the graph (Fig. 1) with nine vertices where each vertex represents a subject and join the vertices by an edge if there is a common student in the subjects they represent.

Construct the degree sequence first, according to the algorithm. It is {7, 6, 6, 6, 6, 6, 6, 6, 3} in the degree sequence. Select the highest degree 7 available for the subject algebra; give it a red hue. Next, select the nonadjacent vertices of the algebra vertex. In this case, the nonadjacent vertex of the algebra vertex is special functions; give it the same red color. The degree sequence then reads as {4, 4, 4, 4, 4, 4, 2}.

Select the next highest degree vertex after that. Select any of the vertices in this diagram that has the same highest degree: advanced complex analysis, differential geometry, operations research, ordinary differential equations, ring and module theory, and partial differential equations. Put "Advanced Complex Analysis" in quotation marks and give it a distinct color, like green. Choose the non-adjacent vertex of the advanced complex analysis vertex; in this case, the non-adjacent vertex is partial differential equations; give it the same green hue; as a result, the degree sequence changes to {3, 2, 2, 2, 1}.

Repeat the procedure, this time assigning a new color—let's say yellow—to the vertex differential geometry, which has the highest degree. Next, choose the vertex that is not adjacent to it, here it is functional analysis, color it by same color, say yellow, The new degree sequence is {1,1,0}.

Following a similar procedure, 4 is the ultimate chromatic number achieved ($\chi(G) = 4$). Exam scheduling is therefore free to occur over four days. Nevertheless, there were vertices with the same greatest degree from which someone could be selected, thus this is not the only solution. Consequently, by choosing more vertices of the highest degree, there may be more than one solution for the same problem.

CONCLUSION

In graph theory, the idea of "graph coloring" refers to the process of giving vertices in a graph distinct colors so that no two adjacent vertices have the same color. This concept can be applied in a variety of real-world scenarios, such as arranging tests or classes so that no two can overlap or take place at the same time. Graph coloring helps make sure that no two conflicting events are planned for the same time slot by visualizing each exam or class as a vertex and each conflict as an edge between vertices, The assignment of distinct frequency bands to transmitters in telecommunications ensures that no transmitter operating in close proximity to another uses the same frequency. Frequency assignment that minimizes interference is made easier with the use of graph coloring. Map coloring: Graph coloring methods make sure that the map uses the fewest colors feasible when coloring a map of nations or regions such that no neighboring regions that share a boundary have the same color.

Additionally, register allocation plays a crucial role in computer science, especially in compiler design. Graph coloring facilitates the efficient and conflict-free allocation of a finite number of CPU registers for variables during execution. Graph coloring offers an organized method for resolving resource allocation or scheduling issues in each of these cases.

Therefore, it is evident that graph theory and its offshoots, the theory of networks and digraphs, have flourished as a field of study as well as methodical tools in operations research. This paper presents an efficient algorithm that solves the Exam Scheduling problems under certain conflicts through the use of graph coloring.

REFERENCES

[1] Angshu Kumar Sinha and Aniruddha Samanta, “Time Scheduling Problem To Make Exam Schedule Using Graph Coloring ”, Advances In Basic And Applied Sciences In Digital Age: Few Selected Areas, PP 57-73.

[2] C. V. R. Harinarayanan and S. Lakshmi, “Applications of Graph Theory in Job Scheduling And Postman’s Problem”, Advances And Applications In Mathematical Sciences, Volume 18, Issue 10, PP 1261-1267, 2019.

[3] Piotr Formanowicz and Krzysztof tanas, “A survey of Graph coloring – its types, methods and applications”, Foundations of computing and Decision Sciences, PP 1-15, 2012.

[4] S. Venu Madhava Sarma, “Applications of Graph Theory In Human Life”, International Journal Of Computer Applications, Vol.1, Issue.02, Pp 21-30, 2012.

[5] Satish Thadani et al, “Applications of Graph coloring in various fields”, Materials today proceedings 66, PP 3498-3501, 2022.

[6] Arifin et al, “Graph Coloring Program of Exam Scheduling Modeling Based on Bitwise Coloring Algorithm Using Python”, Journal of Computer Science 2022, 18 (1): DOI: 10.3844/jcssp.2022.26.32