Chapter 8
Graphs and Recursive Counting

ABSTRACT

The chapter introduces graphs as models for representing objects and relationships, defining concepts such as the Hamiltonian cycle and the Dirac theorem for their analysis. Next, the chapter explores recursive counting, a powerful method that solves complex problems by breaking them into simpler subproblems. This method is applied to the computation of permutations and combinations, highlighting how a recursive approach simplifies the solving of combinatorial problems. The chapter also discusses applications of graphs and recursive counting in various fields, such as network design, planning, and data analysis. Several examples show how graphs and recursive counting can be used to solve combinatorial problems. These examples include the analysis of social platforms, urban road systems, protein interactions, and the simulation of peer-to-peer (P2P) networks. Finally, a problem inspired by the Cretan labyrinth is examined, and several approaches to finding possible paths in a maze are discussed, such as the A algorithm, breadth-first search (BFS), and a recursive approach.*

GRAPHS

In Italy, the shortest line between two points is the arabesque

Ennio Flaiano

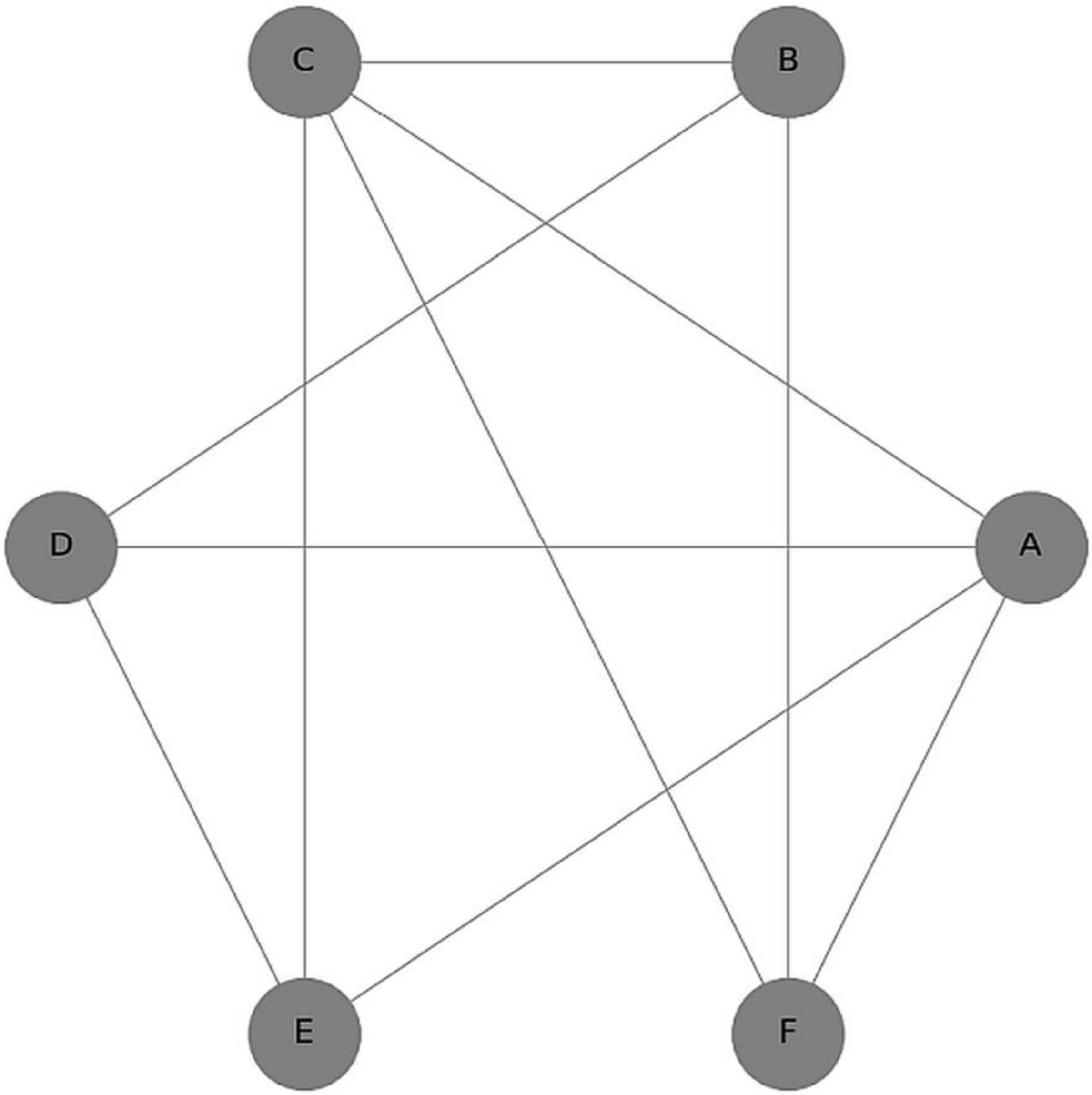
Let us imagine a group of people, and we want to represent the friendships between them. Each person is represented as a *node* (a circle), and every friendship is represented as an *arc* (a line) connecting the nodes. A combinatorial graph is an abstract graph representing a finite collection of objects and their relationships or connections. In a combinatorial graph, edges are defined to satisfy specific rules limiting their structure. Arc analysis is important in combinatorics and involves many concepts and theorems.

Examine the following problem: $2n$ people sit at a circular table; everyone is friends with at least n other person. Can two friends always occupy two seats next to each other? To address this problem, we use graph theory (West, 2001; Chartrand & Zhang, 2013; Gross et al., 2018; Diestel, 2024). We consider each person as a node in a graph that we call G. An arc connects two nodes if the two corresponding people are friends. Then, we create a graph where the nodes represent individuals, and the edges represent the friendship relationships between them. The text states that "everyone is a friend of at least n people". In terms of graphs, this means that each node has a degree of at least n in G. The degree of a node is the number of arcs connected to it. So, each person (node) has at least n friends (connected arcs). If we can find a cycle that visits each node exactly once in our graph, known as a Hamiltonian cycle, then we can answer the question.

We can use Dirac's Theorem to prove that there is a Hamiltonian cycle in our graph: "A graph with n nodes has a Hamiltonian cycle if each node has a degree of at least n/2". We must verify if the condition of Dirac's Theorem is satisfied: is it true that $n \geq (2n)/2$? Yes, n equals $(2n)/2 = n$. Thus, Dirac's Theorem guarantees us that there is a Hamiltonian cycle in graph G. We can arrange people around the circular table following the order of the nodes in this cycle. Each person will sit next to two friends in this arrangement, solving the problem. In conclusion, using graph theory and Dirac's Theorem, we have shown that it is always possible to arrange $2n$ people around a circular table so that two neighbours are friends if each is friends with at least n people.

DOI: 10.4018/979-8-3373-3089-1.ch008

Figure 1. Six friends around a circular table



The graph is produced by a program that simulates the creation of friendships between a group of people and tests whether it is possible to arrange them around a circular table so that each person is seated next to two friends.

Friendships generated:

A: C, F, E, D
B: C, F, D
C: A, B, F, E
D: A, B, E
E: A, D, C