Chapter 4
A Comprehensive Analysis of Stack and Queue Data Structures and Their Uses

S. Rajasekaran
https://orcid.org/0000-0002-7893-9072
University of Technology and Applied Sciences, Ibri, Oman

Mastan Vali Shaik
https://orcid.org/0000-0002-0068-5519
University of Technology and Applied Sciences, Ibri, Oman

ABSTRACT

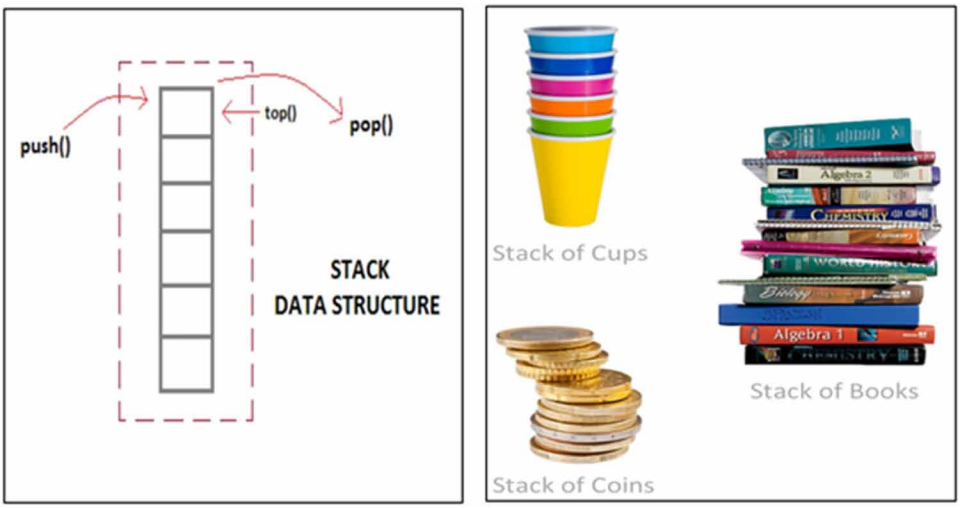
The chapter provides a comprehensive and in-depth exploration of two fundamental linear data structures in computer science: Stack and Queue. It begins with an introduction to these data structures, highlighting their key features and fundamental differences. It then explains various operations that can be performed on both data structures, such as push, pop, enqueue, and dequeue. These implementations are thoroughly compared and contrasted, highlighting their advantages and disadvantages. These applications are discussed in detail, showcasing the versatility of Stack and Queue data structures in different scenarios. The chapter includes Python code snippets demonstrating Stack and Queue data structures further to aid the reader's understanding of the material. This chapter provides a comprehensive and practical guide to Stack and Queue data structures, making it a valuable resource for computer science students, researchers, and practitioners.

DOI: 10.4018/978-1-6684-7100-5.ch004

INTRODUCTION TO STACK

A stack is a simple linear abstract data structure with finite (predefined) capacity that follows the Last In First Out (LIFO) principle. That means the element inserted last into the Stack will be removed first. Conversely, it can say that First In Last Out (FILO). That means the element inserted first into the Stack will be removed last. Both insert and delete operations are performed only on the Top of the Stack. TOP is a marker pointing to the current position in the Stack (Goodrich et al., 2013; Vasudevan et al., 2021). The initial value of Top is -1. Figure 1 provides examples of stack data structure.

Figure 1. Examples of stack data structure



FEATURES OF STACK

The following features characterize the Stack.

1. A Stack consists of an ordered collection of homogeneous (same type) data elements.
2. The Stack accesses its data in LIFO or FILO order.
3. The push() method inserts new items into the Stack, while the drop() function removes components from the Stack. Only at one end of the Stack, termed the Top, are insert and delete operations done.