

Stacks: Advantages and Disadvantages

Advantages:

- LIFO (Last In, First Out) structure suits certain problem-solving approaches.
- Simple implementation.
- Space efficiency in memory allocation.
- Easily reversible operations (pop and push).

Disadvantages:

- Limited access to elements (no random access).
- Dynamic resizing may lead to memory fragmentation.
- Inefficient for searching operations.
- Can lead to stack overflow if the maximum capacity is exceeded.

Real-World Examples:

- **Function Call Management:** Programming languages use stacks to manage function calls, allowing for the execution of functions in the correct order.
- **Expression Evaluation:** Stacks are utilized in evaluating expressions, such as postfix notation, by pushing operands and operators onto the stack and popping them off as needed.
- **Undo Mechanisms:** Text editors, graphic design software, and other applications implement undo functionality using stacks to store previous states.
- **Browser History:** Web browsers utilize stacks to keep track of visited web pages, enabling users to navigate backward and forward through their browsing history.
- **Memory Management:** Operating systems use stacks for managing memory allocation and execution of programs, including managing function calls, local variables, and system resources.