

# Stacks

Joseph Chuang-Chieh Lin (林莊傑)

Department of Computer Science & Engineering,  
National Taiwan Ocean University

Fall 2024



# Outline

- 1 Definition
- 2 Implementation



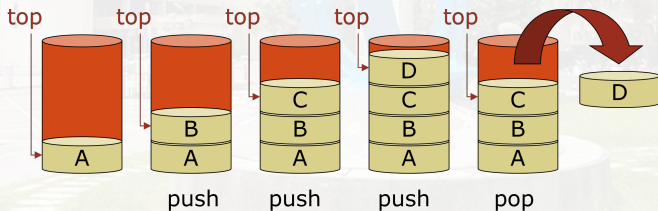
# Outline

- 1 Definition
- 2 Implementation



# Definition

- A stack is an ordered list in which **insertions** and deletions are made at the end “top”.
  - insertions: push/add
  - deletions: pop/remove
- Last-In-First-Out (LIFO).



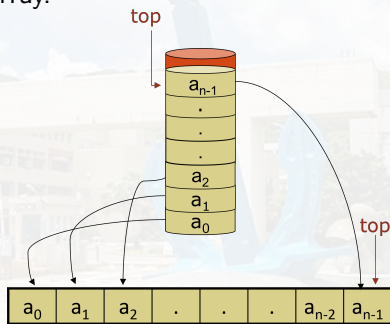
# Outline

- 1 Definition
- 2 Implementation



## Stack Implementation: Array

- The easiest way to implement the stack ADT is using one-dimensional array.

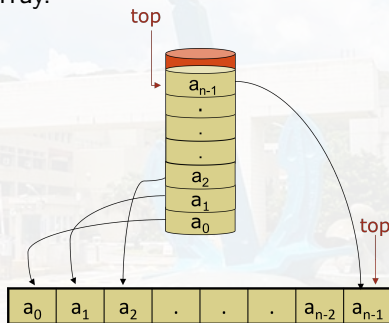


- An example in C++



## Stack Implementation: Array

- The easiest way to implement the stack ADT is using one-dimensional array.



- An example in C++ (another way: using a linked list; will be introduced in the future).



# Functions for Stacks

- Create a stack.
  - Create an empty stack with maximum size `MAX_STACK_SIZE`.

```
#define MAX_STACK_SIZE 101

typedef struct {
    int key; // can be of other types...
    /* other fields? */
} element;

element stack a[MAX_STACK_SIZE];
int top = -1; // initially no element
```





## Functions for Stacks (2/2)

- `IsEmpty`
  - Return `true` if the stack is empty and `false` otherwise.



## Functions for Stacks (2/2)

- **IsEmpty**
  - Return true if the stack is empty and false otherwise.  
 $top < 0$
- **IsFull**
  - Return true if the stack is full and false otherwise.



## Functions for Stacks (2/2)

- **IsEmpty**
  - Return true if the stack is empty and false otherwise.  
`top < 0`
- **IsFull**
  - Return true if the stack is full and false otherwise.  
`top >= MAX_STACK_SIZE-1;`
- **Push (or Add)**
  - Insert the element into the **top** of the stack.



## Functions for Stacks (2/2)

- **IsEmpty**
  - Return true if the stack is empty and false otherwise.  
`top < 0`
- **IsFull**
  - Return true if the stack is full and false otherwise.  
`top >= MAX_STACK_SIZE-1;`
- **Push (or Add)**
  - Insert the element into the **top** of the stack.  
`stack[++top] = element;`
- **Pop (or Delete)**
  - Remove and return the item on the top of the stack.



## Functions for Stacks (2/2)

- IsEmpty

- Return true if the stack is empty and false otherwise.  
`top < 0`

- IsFull

- Return true if the stack is full and false otherwise.  
`top >= MAX_STACK_SIZE-1;`

- Push (or Add)

- Insert the element into the `top` of the stack.  
`stack[++top] = element;`

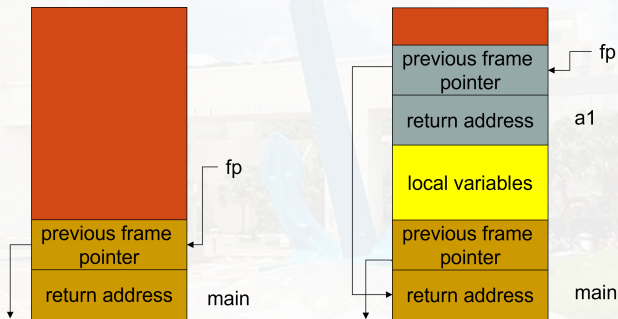
- Pop (or Delete)

- Remove and return the item on the top of the stack.  
`return stack[top--];`



# Supplementary: System Stack

- Stack frame of a function call



system stack **before**  $a_1$  is invoked

system stack **after**  $a_1$  is invoked



# Discussions

