

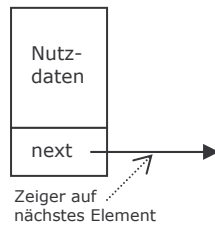


# Übersichtsblatt: Stack

## Definition:

Ein Stack ist eine lineare Datenstruktur, die nach dem Vorbild eines Stapels modelliert ist. Der letzte Gegenstand der auf dem Stapel abgelegt wurde ist der erst Gegenstand den man wieder herunter nehmen muss. Etwas aus der Mitte des Stapels zu nehmen ist nicht möglich.

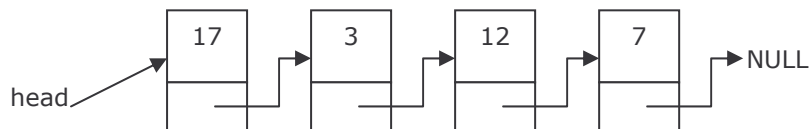
Schematische Darstellung  
eines Stack-Elementes:



C-Implementierung eines  
Stack-Elementes:

```
struct Obj {  
    <typ> data;  
    struct Obj *next;  
}
```

Beispielstack:-



## Operationen:

Aus den oben definierten Eigenschaften eines Stacks ergeben sich 3 Operationen die auf dieser Datenstruktur ausgeführt werden können.

### Push:

Etwas auf dem Stack ablegen.

*Ablauf:*

1. Neues Objekt mit malloc reservieren.
2. Speichern der Daten im neuen Objekt.
3. Einketten als neues Objekt.
4. Den head-Pointer auf das neue Objekt setzen.

*Zeitaufwand:*  $O(1)$

### Pop:

Den obersten Gegenstand vom Stack nehmen.

*Ablauf:*

1. Zeiger auf head-Objekt merken.
2. Head auf Nachfolger setzen.
3. Daten auslesen.
4. Speich des entfernten Objekts freigeben.

*Zeitaufwand:*  $O(1)$

### IsEmpty:

Die Funktion überprüft ob der Stack leer ist.

*Zeitaufwand:*  $O(1)$

