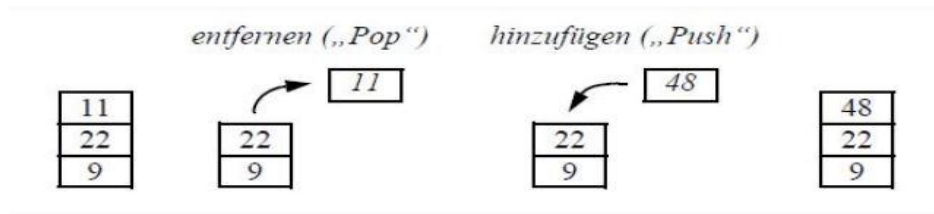


## Aufgaben zu Versuch 1

### Aufgabe 1.1:

Ein Kellerspeicher (Stapel, engl. Stack) ist eine Datenstruktur mit mehreren Einträgen, die nach dem LIFO-Prinzip („Last In First Out“) verwaltet wird: Ein neuer Eintrag wird jeweils oben auf dem Stapel der vorhandenen Einträge abgelegt („Push“-Operation), und es wird beim Lesen jeweils der oberste Eintrag entfernt („Pop“-Operation):



Ein Stack kann durch einen Array realisiert werden. Eine zusätzliche Ganzzahlvariable (der „Stackpointer“) gibt dabei den Arrayindex der Stackspitze an. Die Arrayeinträge **a[0]**, ..., **a[top]** sind also der aktuelle Inhalt des Stacks (**a** ist dabei der Array, **top** der Stackpointer). Eine **Push-Operation** schreibt den neuen Wert nach **a[top+1]** und erhöht **top** um 1; eine **Pop-Operation** liefert den Wert von **a[top]** und senkt **top** um 1. Bei **top==-1** ist der Stack leer.

Schreiben Sie ein Java-Programm, in dem eine Klasse für solche Stacks definiert und benutzt wird. Objekte der Klasse sollen folgende Eigenschaften haben:

- Attribute sind, wie oben beschrieben, ein Array mit **int**-Einträgen und eine **int**-Variable für den Stackpointer. Die Attribute sollen von außen nicht direkt zugreifbar sein.
- Drei Methoden sollen Folgendes leisten:
  - Die **Push-Methode** legt einen Wert auf dem Stack ab, sofern er nicht voll ist. Sie erhält den abzulegenden Wert als Parameter und liefert als Rückgabewert eine **0** bei erfolgreicher Operation oder eine **-1**, wenn der Stack bereits voll ist.
  - Die **Pop-Methode** entfernt den obersten Wert vom Stack, sofern er nicht leer ist. Sie liefert als Rückgabe den gelesenen Wert oder eine **-1**, wenn der Stack leer ist.
  - Die **Print-Methode** gibt den Stackinhalt auf den Bildschirm aus.
- Ein **Konstruktor**: Der Konstruktor soll als Parameterwert die gewünschte Arraygröße übergeben bekommen. Er soll den Array erzeugen und den Stackpointer so vorbesetzen, dass der Stack zu Beginn leer ist.

Das Hauptprogramm soll zunächst ein **Stack-Objekt** erzeugen. Dabei soll die Stackgröße über die Tastatur eingelesen und dann an den Konstruktor übergeben werden. Anschließend soll der Benutzer die Möglichkeit haben, eine der drei Methoden auszuwählen. Das Hauptprogramm soll, falls nötig, von der Tastatur den Parameterwert für die Methode einlesen, die Methode damit aufrufen und den Rückgabewert der Methode auf den Bildschirm ausgeben.

Dieser Vorgang soll beliebig oft ausgeführt werden, bis der Benutzer eine Beendigung wünscht.

### **Zu beachten:**

- Alle Tastatureingaben sollen im Hauptprogramm vorgenommen werden, **nicht** in den Objektmethoden selbst. Auch sollen innerhalb der **Push**- und der **Pop**-Operation keine Bildschirmausgaben erfolgen.
- **Bei der Programmierung dürfen Sie nicht auf die vordefinierte Java-Klasse „Stack“ oder ähnliche Klassen zurückgreifen!**

### **Aufgabe 1.2:**

Definieren Sie nun eine Unterklasse (also eine abgeleitete Klasse) Ihrer Klasse aus Aufgabe 1.1.

Diese Unterklasse soll zusätzlich die folgenden Methoden bieten:

- Einen Konstruktor, der neben der gewünschten Arraygröße einen Ganzzahlwert **wert** übergeben bekommt und den neu erzeugten Stack mit den Werten **wert, wert+1, wert+2, ...** vollständig füllt. Er soll zunächst zur Erzeugung des Arrays den Konstruktor der Oberklasse aufrufen und dann selbst die Werte in den Stack schreiben.
- Eine Methode, die genau dann **true** als Rückgabewert liefert, wenn der Stack leer ist.
- Eine erweiterte **Push**-Methode, die einen Array übergeben bekommt und alle Arrayeinträge auf dem Stack ablegt. Dies soll nur geschehen, wenn der Stack noch genügend Platz für alle diese neuen Einträge hat, sonst soll sein Inhalt unverändert bleiben. Rückgabewert soll, wie bei der einfachen **Push**-Methode, **0** bzw. **-1** sein.

Ändern Sie Ihr Hauptprogramm so, dass ein **Stack**-Objekt der Unterklasse erzeugt wird. Der Stack soll die Größe **6** haben und mit der Wertfolge **10 11 12 13 14 15** vorbesetzt werden. Wie in Aufgabe 1 soll es möglich sein, aus dem Hauptprogramm die Methoden der Unterklasse (einschließlich derer der Oberklasse) wiederholt aufzurufen.