AuD Zettel 04

Aufgabe 2 a)

```
void stackArrayInitialize(int n):
    setze stack1 = 0;          // Einfüge Zeiger
    setze stack2 = n;          // Zeiger des obersten Elements
    setze int[] array = new int[n];   (Initialisierung auch als Felder)

int[] push(int stackID, int value):
    Falls (stack1 + 1 >= stack2)   // Stacks zu groß
        return array;
    Falls (stackID == 1)   // Stack1 soll vergrößert werden
        setze array[stack1] = value;
        stack1++;
        return array;
    sonst                  // stack2 soll vergrößert werden
        stack2--;
        setze array[stack2] = value;
        return array;

int[] pop(int stackID)
    Falls (stackID == 1)
        Falls (stack1 == 0)   // stack1 leer?
            return array;
        sonst
            Lösche value in array[stack1-1];
            stack1--;
            return array;   -> bzw. array.length 0
    sonst
        Falls (stack2 == n)   // stack2 leer?
            return array;
        sonst
            Lösche value in array[stack2];
            stack2++;  return array;
```

BRUNNEN

Das Verfahren, welches hier verwendet wird, ist recht simpel. Man initialisiert ein Array der Größe n und lässt beide Stacks aufeinander zu wachsen. D.h. ein Stack beginnt bei Index 0 und der zweite bei n-1. Beim größer werden, laufen sie jeweils richtung Mitte des Arrays.

6)