

Brug af variabler i C#

Af Michell Cronberg (michell@cronberg.dk)

Arrays

Arrays (System.Array) er en referencebaseret type hvorfor data placeres på heap'en.


Her erklæres en variabel som kan pege på et int-array:

```
int[] a;
```

Stack	Heap
a	

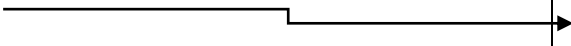
Så oprettes et array, og variabel og array bindes sammen:

```
int[] a;  
a = new int[3];
```

Stack	Heap		
a = 	0	1	2
	0	0	0

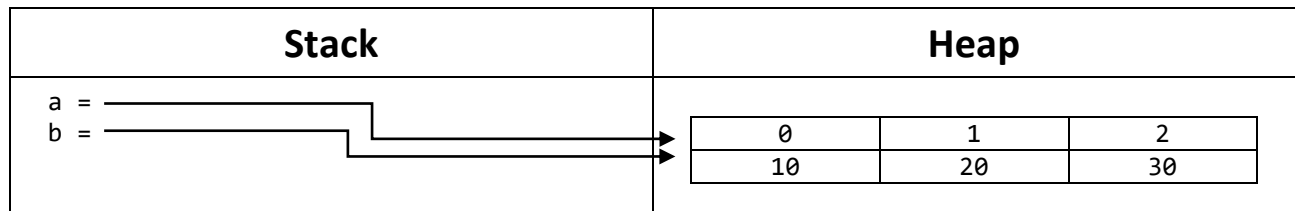
Og værdier tildeles:

```
int[] a;  
a = new int[3];  
a[0] = 10;  
a[1] = 20;  
a[2] = 30;
```

Stack	Heap		
a = 	0	1	2
	10	20	30

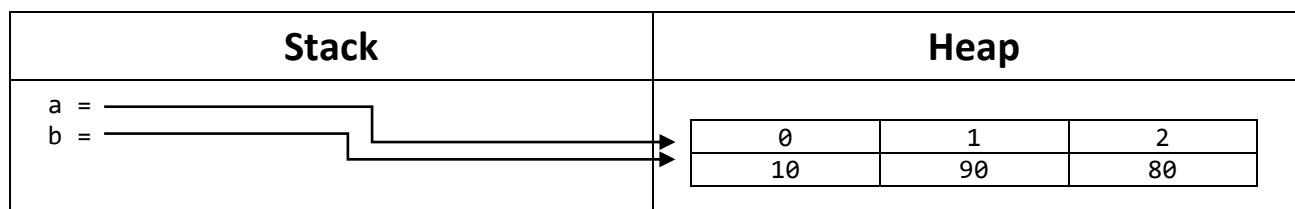
Her erklæres en ny variabel som kan pege på et int-array, og referencen fra det tidligere arrays kopieres:

```
int[] a;  
a = new int[3];  
a[0] = 10;  
a[1] = 20;  
a[2] = 30;  
int[] b;  
b = a;
```



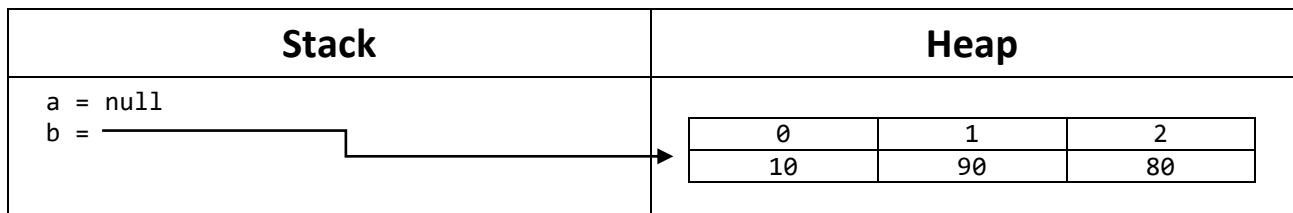
Nu peger både a og b på samme array, og det er underordnet om den ene eller anden variabel benyttes:

```
int[] a;  
a = new int[3];  
a[0] = 10;  
a[1] = 20;  
a[2] = 30;  
int[] b;  
b = a;  
b[1] = 90;  
a[2] = 80;
```



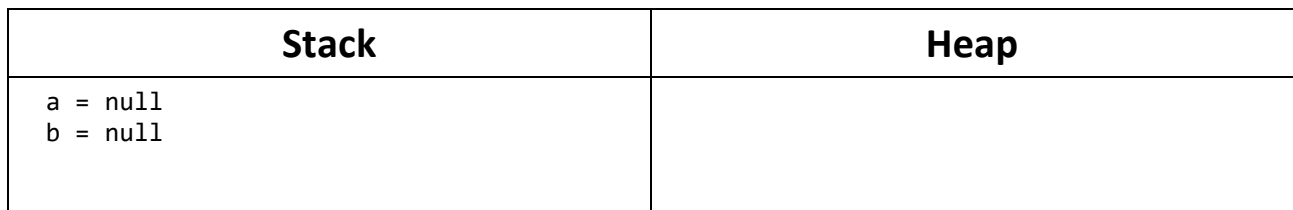
Så længe et eller andet peger på array'et forbliver det på heap'en:

```
int[] a;  
a = new int[3];  
a[0] = 10;  
a[1] = 20;  
a[2] = 30;  
int[] b;  
b = a;  
b[1] = 90;  
a[2] = 80;  
a = null;
```



Men hvis alle referencer er væk bliver array'et fjernet og pladsen frigivet til anden brug:

```
int[] a;
a = new int[3];
a[0] = 10;
a[1] = 20;
a[2] = 30;
int[] b;
b = a;
b[1] = 90;
a[2] = 80;
a = null;
b = null;
```



Men variablerne a og b kan jo stadig benyttes til et eventuelt nyt int-array.