

Brug af variabler i C#

Af Michell Cronberg (michell@cronberg.dk)

Brug af værdibaserede typer

Når der er tale om simple værdibaserede typer (int, double, bool, DateTime, struct med videre) placeres både variabler og tilhørende data i den stack-relaterede hukommelse.

Eksempel på brug af heltal

Her et eksempel på erklæring af to heltal:

```
int i = 1;  
int x = 2;
```

Det resulterer i følgende diagram:

Stack	Heap
i = 1 x = 2	

Når den ene variabel tildeles indholdet af den anden variabel kopieres den konkrete værdi. Her sættes x til at være lig med værdien af i:

```
int i = 1;  
int x = 2;  
x = i;
```

Nu ser hukommelsen således ud:

Stack	Heap
i = 1 x = 1	

Når værdien af i ændres vil det ikke påvirke værdien af x fordi der udelukkende arbejdes med værdier:

```
int i = 1;  
int x = 2;  
x = i;  
i = 3;
```

Variablen x har forsat værdien 1:

Stack	Heap
i = 3 x = 1	

Eksempel på brug af en struct

Her er et andet eksempel baseret på en struct (som jo også er en værdibaseret type), der ser således ud:

```
struct Person {  
    public int Id;  
    public string Navn;  
}
```

Følgende kode:

```
Person p1 = new Person() { Id = 1, Navn = "Mikkel" };  
Person p2 = new Person() { Id = 2, Navn = "Mathias" };
```

resulterer således i dette diagram:

Stack	Heap
p1 = { Id: 1, Navn: "Mikkel"} p2 = { Id: 2, Navn: "Mathias"}	

Bemærk, at data er placeret direkte på stack'en fordi der er tale om en struct.

Hvis p1 sættes lig med p2:

```
Person p1 = new Person() { Id = 1, Navn = "Mikkel" };  
Person p2 = new Person() { Id = 2, Navn = "Mathias" };  
p1 = p2;
```

vil det udelukkende resultere i en kopiering af data direkte på stack'en:

Stack	Heap
p1 = { Id: 2, Navn: "Mathias"} p2 = { Id: 2, Navn: "Mathias"}	

og når p2 tilrettes efterfølgende:

```
Person p1 = new Person() { Id = 1, Navn = "Mikkel" };  
Person p2 = new Person() { Id = 2, Navn = "Mathias" };  
p1 = p2;  
p2.Navn = "Lene";
```

vil det ikke have nogen som helst konsekvens på data i p1:

Stack	Heap
p1 = { Id: 2, Navn: "Mathias"} p2 = { Id: 2, Navn: "Lene"}	