

1.) Aufgabenblatt - Parallele Programmierung



Kompetenzen ▾

Thread: Threadsynchronisation, Semaphoren

1.) Beispiel - Threadsynchronisation (10.Punkte)

In einer Fabrik werden Werkstücke hergestellt. Die Bearbeitungsanlage besteht dabei aus einem **Kran** einer **Maschine A** und einer **Maschine B**.

Der Bearbeitungsprozess für Werkstücke durchläuft folgende Schritte:

► Definition: Prozessablauf ▾

- Ein Werkstück wird vom **Kran** von Lager1 zu **MaschineA** transportiert.
- Das Werkstück wird von **MaschineA** bearbeitet. Dazu wird die **Process()** Methode von **MaschineA** aufgerufen.
- Das Werkstück wird anschließend vom **Kran** von **MaschineA** zu **MaschineB** transportiert.
- Das Werkstück wird nun von **MaschineB** bearbeitet. Dazu wird die **Process()** Methode von **MaschineB** aufgerufen.
- Zuletzt wird das Werkstück vom **Kran** von **MaschineB** ins Lager2 transportiert.



Hinweis: Führen Sie **MaschineA**, **MaschineB** und den **Kran** jeweils in einem eigenen Thread aus. Synchronisiere Sie die einzelnen Threads um den geforderten Prozessablauf abzubilden.

► Codebeispiel: Programm.cs ▾

```

1  // -----
2  // Prozessablauf:
3  // -----
4  // 1.Step: Crane->Move("Storage", "MachineA")
5  // 2.Step: MachineA->Process()
6  // 3.Step: Crane->Move("MachineA", "MachineB")
7  // 4.Step: MachineB->Process()
8  // 5.Step: Crane->Move("MachineB", "Storage")
9
10 // -----
11 // Klasse: MachineA
12 // -----
13
14 public class MachineA {
15
16     public void Run(){
17         while(true){
18             ...
19             Process();
20             ...
21         }
22     }
23
24     private void Process(){
25         Thread.Sleep(100);
26         Console.WriteLine(
27             "MachineA: finished work"
28         );
29     }
30 }
31
32 // -----
33 // Klasse: MachineB
34 // -----
35 public class MachineB {
36
37     public void Run(){
38         while(true){
39             ...
40             Process();
41             ...
42         }
43     }
44
45     private void Process(){
46         Thread.Sleep(150);
47         Console.WriteLine(
48             "MachineB: finished work"
49         );
50     }
51 }
52

```

```
53 // -----
54 // Klasse: Crane
55 // -----
56 public class Crane {
57
58     public void Run(){
59         while(true){
60             ...
61             Move("Storage", "MachineA");
62             ...
63             Move("MachineA", "MachineB");
64             ...
65             Move("MachineB", "Storage");
66             ...
67         }
68     }
69
70     private void Move(
71         string from, string to
72     ){
73         Thread.Sleep(200);
74         Console.WriteLine(
75             $"moving from {from} to {to}"
76         );
77     }
78 }
```

