

Letzte Chance!!!

- Exceptions
- Thread, Runnable
- Synchronized
- Wait, notify, notifyAll
- Thread States
- Semaphoren
- JCSP
- Swing
- JOMP
- Linearizability
- History
- Amdahl's Law
- Atomic

Danke an Jeremia

Java	[DIM=800,threads=1]:	2690 ms
C++	[DIM=800,threads=1]:	1706 ms
Java	[DIM=800,threads=4]:	760 ms
C++	[DIM=800,threads=4]:	432 ms
Java	[DIM=1000,threads=1]:	5265 ms
C++	[DIM=1000,threads=1]:	4423 ms
Java	[DIM=1000,threads=4]:	1570 ms
C++	[DIM=1000,threads=4]:	1114 ms
Java	[DIM=1100,threads=1]:	7714 ms
C++	[DIM=1100,threads=1]:	6352 ms
Java	[DIM=1100,threads=4]:	2291 ms
C++	[DIM=1100,threads=4]:	1599 ms
Java	[DIM=1500,threads=1]:	28747 ms
C++	[DIM=1500,threads=1]:	19021 ms
Java	[DIM=1500,threads=4]:	7932 ms
C++	[DIM=1500,threads=4]:	5004 ms
Java	[DIM=2000,threads=1]:	68808 ms
C++	[DIM=2000,threads=1]:	43657 ms
Java	[DIM=2000,threads=4]:	18648 ms
C++	[DIM=2000,threads=4]:	11104 ms





Wie verhält sich folgendes Programm?

```
boolean b = true;  
while(b = true)  
    b = false;
```

Compiler Fehler

Kein Schleifendurchgang

Ein einziger Schleifendurchgang

Läuft unendlich lange



Wie verhält sich folgendes Programm?

```
boolean b = true;  
while(b = true)  
    b = false;
```

Compiler Fehler

Kein Schleifendurchgang

Ein einziger Schleifendurchgang

Läuft unendlich lange



Welchen Typ hat der Rückgabewert von `Channel.one2one().in()`?

`org.jcsp.lang.Guard`

`org.jcsp.lang.ChannelOutput`

`java.io.InputStream`

`org.jcsp.lang.CSProcess`



Welchen Typ hat der Rückgabewert von `Channel.one2one().in()`?

`org.jcsp.lang.Guard`

`org.jcsp.lang.ChannelOutput`

`java.io.InputStream`

`org.jcsp.lang.CSProcess`



Welches dieser Begriffe ist kein Keyword von OMP?

critical

single

multiple

schedule



Welches dieser Begriffe ist kein Keyword von OMP?

critical

single

multiple

schedule



Wo gibt es einen Compiler Fehler?

```
1: Runnable r = new Runnable() {public void run(){} };  
2: try { r.start();  
3: } finally {  
4: throw new NullPointerException(); }
```

Zeile 1

Zeile 2

Zeile 3

Zeile 4



Wo gibt es einen Compiler Fehler?

```
1: Runnable r = new Runnable() {public void run(){} };  
2: try { r.start();  
3: } finally {  
4: throw new NullPointerException(); }
```

Zeile 1

Zeile 2

Zeile 3

Zeile 4



Was ist das Prinzip des MVC Patterns?

Die Verwendung von Listener, um innerhalb des Programms Events weiterzugeben.

Das Model unabhängig vom GUI zu implementieren.

Niemals `getGraphics()` auf einer Swing Komponente aufzurufen, sondern alles in der Methode `paint(g)` zu zeichnen.

Die Synchronisation zwischen Threads und dem EDT mittels `SwingUtilities.invokeLaterAndWait(...)`.



Was ist das Prinzip des MVC Patterns?

Die Verwendung von Listener, um innerhalb des Programms Events weiterzugeben.

Das Model unabhängig vom GUI zu implementieren.

Niemals `getGraphics()` auf einer Swing Komponente aufzurufen, sondern alles in der Methode `paint(g)` zu zeichnen.

Die Synchronisation zwischen Threads und dem EDT mittels `SwingUtilities.invokeLaterAndWait(...)`.



Was führt zu einer Exception?

```
synchronized(lock) {  
    synchronized(lock) {  
        lock.wait();  
    }  
}
```

notify() auf einem Objekt aufzurufen, auf dem noch kein wait() aufgerufen wurde.

notifyAll() auf einem nicht-synchronisierten Objekt aufzurufen.

```
synchronized(lock) {  
    lock.wait();  
    lock.wait();  
}
```



Was führt zu einer Exception?

```
synchronized(lock) {  
    synchronized(lock) {  
        lock.wait();  
    }  
}
```

notify() auf einem Objekt aufzurufen, auf dem noch kein wait() aufgerufen wurde.

notifyAll() auf einem nicht-synchronisierten Objekt aufzurufen.

```
synchronized(lock) {  
    lock.wait();  
    lock.wait();  
}
```



Was ist der Output dieses Programm?

```
AtomicInteger i = new AtomicInteger(0);  
i.getAndSet(i.getAndIncrement() + i.getAndAdd(i.getAndDecrement()));  
System.out.println(i.get());
```

-1

0

1

2



Was ist der Output dieses Programm?

```
AtomicInteger i = new AtomicInteger(0);  
i.getAndSet(i.getAndIncrement() + i.getAndAdd(i.getAndDecrement()));  
System.out.println(i.get());
```

-1

0

1

2



```
i.getAndSet(i.getAndIncrement() + i.getAndAdd(i.getAndDecrement()));
```

```
i.getAndIncrement() + i.getAndAdd(i.getAndDecrement())
```

```
=> ret=0, i=1
```

```
i.getAndAdd(i.getAndDecrement())
```

```
=> ret=1 , i=0
```

```
i.getAndAdd(1)
```

```
=> ret=0 , i=1
```

```
i.getAndSet(0 + 0);
```

```
=> ret=1, i=0
```

```
==> i.get() = 0
```



Welchen Methoden in Java entsprechen die Methoden P() und V() aus der Semaphoren Theorie?

P = ask(), V = respond()

P = demand(), V = restore()

P = power(), V = want()

P = acquire(), V = release()



Welchen Methoden in Java entsprechen die Methoden P() und V() aus der Semaphoren Theorie?

P = ask(), V = respond()

P = demand(), V = restore()

P = power(), V = want()

P = acquire(), V = release()



Wie verhält sich folgendes Programm?

```
t = new Thread() {  
    public void run() {  
        t.start();  
    } };  
t.start();
```

NullPointerException

IllegalThreadStateException

Startet zwei neue Threads

Startet unendlich viele Threads



Wie verhält sich folgendes Programm?

```
t = new Thread() {  
    public void run() {  
        t.start();  
    } };  
t.start();
```

NullPointerException

IllegalThreadStateException

Startet zwei neue Threads

Startet unendlich viele Threads



Wie viele Zeilen hat die Klasse Thread?





Wie viele Zeilen hat die Klasse Thread?

1841

Vielen Dank 😊

