

Nachbau der Turing-Maschine

Beispiel: Dekrementierung einer Binärzahl

Folgendes Programm ist für die Turing-Maschine gegeben:

Aktueller Zustand	Gelesenes Zeichen	Geschriebenes Zeichen	Bewegung	Neuer Zustand
S	#	#	LEFT	S
S	1	0	RIGHT	R
S	0	1	LEFT	L
R	0	0	RIGHT	R
R	1	1	RIGHT	R
R	#	#	LEFT	W
W	1	1	RIGHT	HALT
W	0	0	RIGHT	HALT
W	#	#	RIGHT	HALT

Aktueller Zustand	Gelesenes Zeichen	Geschriebenes Zeichen	Bewegung	Neuer Zustand
L	0	1	LEFT	L
L	1	0	RIGHT	R
L	#	#	RIGHT	R

Die ersten drei Einzelschritte erklärt:

Das Band wird initialisiert mit:

```
# 1 1 0 0 0 {#}
```

Das Vorbelegungszeichen auf dem Band ist # und die Maschine befindet sich im Zustand S.

Schritt 1: Die TM liest das Zeichen # an der Position des S/L-Kopfes vom Band. Der S/L-Kopf schreibt **keine Änderung**, bewegt sich nach **links** und die TM bleibt im Zustand S.

Schritt 2: Die TM liest das Zeichen 0 an der Position des S/L-Kopfes vom Band. Der S/L-Kopf schreibt das Zeichen 1, bewegt sich nach **links** und die TM wechselt in Zustand L.

Schritt 3: Die TM liest das Zeichen 0 an der Position des S/L-Kopfes vom Band. Der S/L-Kopf schreibt das Zeichen 1, bewegt sich nach **links** und die TM wechselt in Zustand L.

Alle Schritte in der jshell:

```
jshell> turingMachine.run()
```

```
0:  # 1 1 0 0 0 {#}  -- S
1:  # 1 1 0 0 {0} #  -- S
2:  # 1 1 0 {0} 1 #  -- L
3:  # 1 1 {0} 1 1 #  -- L
4:  # 1 {1} 1 1 1 #  -- L
5:  # 1 0 {1} 1 1 #  -- R
6:  # 1 0 1 {1} 1 #  -- R
7:  # 1 0 1 1 {1} #  -- R
8:  # 1 0 1 1 1 {#}  -- R
9:  # 1 0 1 1 {1} #  -- W
10: # 1 0 1 1 1 {#}  -- HALT
```

Beispiel: Einsen nach rechts schieben

```
jshell> turingMachine.run()
0:  S 0 1 0 1 0 {1} S  -- S
1:  S 0 1 0 1 {0} 1 S  -- S
2:  S 0 1 0 {1} 0 1 S  -- 0
3:  S 0 1 0 0 {0} 1 S  -- 1
4:  S 0 1 0 0 0 {1} S  -- 1
5:  S 0 1 0 0 {0} 1 S  -- D
6:  S 0 1 0 {0} 1 1 S  -- S
7:  S 0 1 {0} 0 1 1 S  -- 0
8:  S 0 {1} 0 0 1 1 S  -- 0
9:  S 0 0 {0} 0 1 1 S  -- 1
10: S 0 0 0 {0} 1 1 S  -- 1
11: S 0 0 0 0 {1} 1 S  -- 1
12: S 0 0 0 {0} 1 1 S  -- D
13: S 0 0 {0} 1 1 1 S  -- S
14: S 0 {0} 0 1 1 1 S  -- 0
```

```
15: S {0} 0 0 1 1 1 S -- 0
16: {S} 0 0 0 1 1 1 S -- 0
17: S {0} 0 0 1 1 1 S -- HALT
```

Um diese Ausgaben auch in der LiveView zu erzeugen, führe diese Kommandos in der jshell aus:

Initialisiert das Band:

```
String initialContent = "#11000#"
```

oder 2. Beispiel:

```
String initialContent = "S010101S"
```

Vorbelegungszeichen setzen:

```
int startPosition = initialContent.length() - 1
```

oder 2. Beispiel:

```
int startPosition = initialContent.length() - 2
```

Der Turing-Maschine das Band, den Zustand, das Vorbelegungszeichen und die Tabelle übergeben:

```
TM turingMachine = new TM(initialContent, "S", startPosition, "decrement");
```

oder 2. Beispiel:

```
TM turingMachine = new TM(initialContent, "S", startPosition, "moveOnes");
```

Turing-Maschine starten:

```
turingMachine.run()
```

Hier die Ausgabe:

#	1	1	0	0	0	{#}	-- S
#	1	1	0	0	0	{#}	-- S
#	1	1	0	0	{0}	#	-- S
#	1	1	0	{0}	1	#	-- L
#	1	1	{0}	1	1	#	-- L
#	1	{1}	1	1	1	#	-- L
#	1	0	{1}	1	1	#	-- R
#	1	0	1	{1}	1	#	-- R
#	1	0	1	1	{1}	#	-- R
#	1	0	1	1	1	{#}	-- R
#	1	0	1	1	{1}	#	-- W
#	1	0	1	1	1	{#}	-- HALT

