



## Spiel und Spaß mit Turingmaschinen

1. Entwerfe eine Turingmaschine, die abwechselnd Nullen und Einsen aufs Band schreibt.

*Beispiel:*  $\square \rightsquigarrow \boxed{0} \boxed{1} \boxed{0} \boxed{1} \boxed{0} \boxed{1} \dots$

2. Entwerfe eine Turingmaschine, die eine 0 vor die gesamte Eingabe setzt und dabei die Eingabe nach rechts verschiebt.

*Beispiel:*  $\boxed{1} \boxed{0} \boxed{1} \boxed{1} \square \rightsquigarrow \boxed{0} \boxed{1} \boxed{0} \boxed{1} \boxed{1} \square$

3. Entwerfe eine Turingmaschine, die eine gegebene natürliche Zahl um Eins erhöht. Die Eingabe erfolgt als Binärdarstellung (mit den Einern vorne, den Zweiern an zweiter Stelle, den Vierern dahinter – also genau umgekehrt als in der üblichen Notation).

*Beispiele:*  $\boxed{0} \boxed{1} \boxed{1} \square \rightsquigarrow \boxed{1} \boxed{1} \boxed{1} \square$ ,  $\boxed{1} \boxed{1} \boxed{1} \square \rightsquigarrow \boxed{0} \boxed{0} \boxed{0} \boxed{1} \square$

*Bonusaufgabe:* Schreibe die Maschine so um, dass Ein- und Ausgabe in der gewohnten Schreib- und Leserichtung erfolgen (also mit den Einern ganz rechts statt ganz links). Beachte, dass die Turingmaschine trotzdem am linken Ende startet!

4. Entwerfe eine Turingmaschine, die die Eingabe vollständig kopiert.

*Beispiel:*  $\boxed{1} \boxed{0} \boxed{1} \square \rightsquigarrow \boxed{1} \boxed{0} \boxed{1} \square \boxed{1} \boxed{0} \boxed{1} \square$

5. Entwerfe eine Turingmaschine, die prüft, ob die Eingabe ein Palindrom ist (zum Beispiel ist „0010100“ ein Palindrom, aber „0010110“ nicht). Meinetwegen darf dabei die Eingabe zerstört werden. Die Maschine soll je nachdem, ob die Eingabe ein Palindrom ist, in einem anderen Zustand halten.

6. Entwerfe eine Turingmaschine, die eine Eins, dann zwei Einsen, dann drei Einsen, und so weiter aufs Band schreibt.

*Beispiel:*  $\square \rightsquigarrow \boxed{1} \square \boxed{1} \boxed{1} \square \boxed{1} \boxed{1} \boxed{1} \square \boxed{1} \boxed{1} \boxed{1} \boxed{1} \square \boxed{1} \boxed{1} \boxed{1} \boxed{1} \boxed{1} \square \dots$

7. Entwerfe eine Turingmaschine, die vergleicht, ob zwei Ziffernfolgen gleich sind oder nicht. Sie soll in verschiedenen Zuständen halten, je nachdem, ob die Folgen gleich sind oder nicht.

*Beispiel:* Bei Eingabe von 

0	1	1	0	0	1	0	0
---	---	---	---	---	---	---	---

 soll die Turingmaschine in einem anderen Zustand halten als bei 

0	1	1	0	0	1	0	1
---	---	---	---	---	---	---	---

.

8. Entwerfe eine Turingmaschine, die zwei in Binärdarstellung gegebene Zahlen addiert.

*Beispiele:*

1	0	1	0	1	1	1
---	---	---	---	---	---	---

 $\rightsquigarrow$ 

1	1	0	0	1
---	---	---	---	---

9. Entwerfe eine Turingmaschine, die das gesamte Band (nach rechts hin) mit Nullen und Einsen vollschreibt: Und zwar soll an die  $n$ -te Position genau dann eine Eins geschrieben werden, wenn  $n$  eine Primzahl ist.

*Beispiel:*

--

 $\rightsquigarrow$ 

0	1	1	0	1	0	1	0	0	0	1	0	1	0	0	0	1	0	
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	--

...

10. Es macht keinen wesentlichen Unterschied, ob das Band in beide Richtungen beliebig lang ist oder nur nach rechts. Um das einzusehen, erkläre wie man aus einer Turingmaschine, die ein beidseitig unbeschränktes Band erwartet, eine Maschine basteln kann, die auch auf einem nur nach rechts beliebig langen Band funktioniert.