

Aufgabe 1:

1. Geben Sie einen regulären Ausdruck an, der die folgende Sprache darstellt:

$$L = \{w \in \Sigma_{\text{Bool}}^* \mid \text{in } w \text{ kommt nach jeder 1 mindestens eine 0}\}$$

2. Geben Sie einen regulären Ausdruck an, der die Menge der akzeptierten Wörter des folgenden Automaten darstellt.

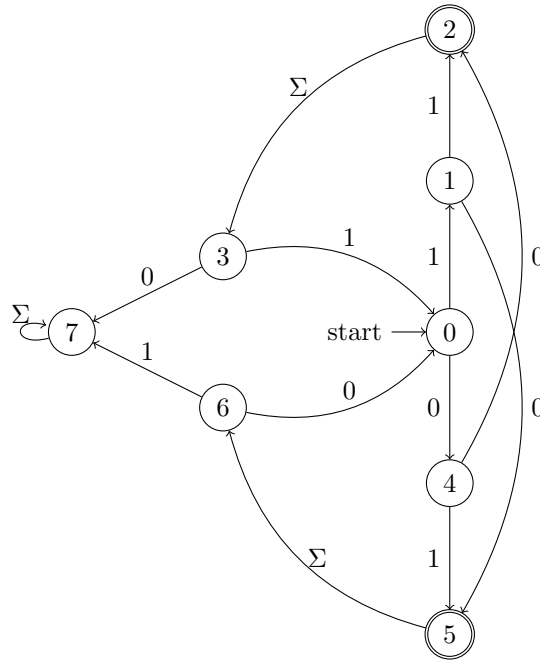


Figure 1: DEA für Aufgabe 1.1.

3. Betrachten Sie folgende Funktionen

$$g_1(n) = \frac{n}{\sqrt{n}}, \quad g_2(n) = 10 \cdot \log(n), \quad g_3(n) = \frac{1}{4}n \quad \text{und} \quad g_4(n) = \frac{1}{10} \cdot \sum_{i=1}^n i.$$

Geben Sie für jede Funktion g_i eine kleinstmögliche Funktion f_i , sodass $g_i \in \mathcal{O}(f_i)$ gilt. Geben Sie zusätzlich kleinstmögliche Werte für die Konstante c , sodass die Bedingung aus der Definition mit $n_0 = 1$ erfüllt ist.

4. Sortieren Sie die Funktionen g_i , mit $i \in \{1, 2, 3, 4\}$, aus Aufgabe 1.3, nach ihrem asymptotischen Wachstum. Begründen Sie kurz Ihre Antwort.
5. Geben Sie für folgende Programmausschnitte die Komplexität in Abhängigkeit von n an:

- ```
int counter = 0;
for (int i=0; i<n; i++){
 for (int k=0; k < i/2; k++){
 counter++;
 }
}
```
- ```
int counter = 0;
for (int i=0; i<n; i++){
    for (int k=0; k < Math.sqrt(n); k++){
        counter++;
    }
}
```
- ```
public static int rek(int n){
 if (n<=1){
 return 1;
 }
 return rek(n/2);
}
```

### Aufgabe 2:

Zahlreiche Programmiersprachen (z.B. Perl, Python, JavaScript) und Unix-Werkzeuge (z.B. bash, grep) arbeiten mit einer erweiterten Syntax für reguläre Ausdrücke.

1. Verschaffen Sie sich einen Überblick über die Funktionsweise dieser Ausdrücke. Hierbei könnten Ihnen z.B. die folgenden Web-Sites helfen: RegEx 101.  
Beachten Sie, dass sich die Syntax der regulären Ausdrücke zu denen aus der Vorlesung unterscheiden. Das + ist zum Beispiel ein |.
2. Formulieren Sie dann möglichst kurze reguläre Ausdrücke für die folgenden Sprachen:

(a)  $L_1 = \{w \in \Sigma_{Digit}^* \mid |w|_0 = 2\}$   
mit  $\Sigma_{Digit} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

(b)  $L_2 = \{w \in \Sigma_{Bool}^* \mid w \text{ enthält das Teil-wort } 0101\}$

(c)  $L_3 = \{ w \in \Sigma_{German}^* \mid w \text{ ist ein deutscher Satz;}$   
d.h.  $w$  beginnt mit einem Grossbuchstaben und endet mit einem Punkt}  
wobei  $\Sigma_{German} = \{A, \dots, Z, a, \dots, z, 0, \dots, 9\} \cup \{ , \} \cup \{ . \} \cup \{ ; \} \cup \{ \_ \}$

**Anmerkung:**  $\_$  steht für ein Leerzeichen.

3. Nennen Sie zwei Bedingungen, die ein beliebiges  $w \in \Sigma_{Bool}^*$  erfüllen muss, um vom regulären Ausdruck  $(10|01)^*$  erkannt zu werden.