

PD Dr. Jan Johannsen
 Elisabeth Lempa
 Luca Maio

Ludwig-Maximilians-Universität München
 Institut für Informatik
 Besprechung 25.06.2026 bis 29.06.2026
 Abgabe bis 07.07.2026, 14:00 Uhr

Lösungsvorschlag zur Klausurvorbereitungsaufgabe zur Übung 9 zur
 Vorlesung

Formale Sprachen und Komplexität

Wenn Sie Automaten angeben, tun Sie dies immer in Form eines Zustandsgraphen. Andere Formen der Darstellung (z.B. als Liste von Übergängen) werden nicht gewertet, da sie sehr viel aufwändiger zu korrigieren sind. Vergessen Sie nicht, im Zustandsgraph Start- und Endzustände zu markieren.

Erlaubte Konstrukte für

LOOP-Programme	WHILE-Programme	GOTO-Programme
$x_i := x_j + c$	$x_i := x_j + c$	$M_k : x_i := x_j + c$
$x_i := x_j - c$	$x_i := x_j - c$	$M_k : x_i := x_j - c$
$x_i := c$	$x_i := c$	$M_k : x_i := c$
$P_1; P_2$	$P_1; P_2$	$P_1; P_2$
LOOP x_i DO P END	LOOP x_i DO P END	$M_k : \mathbf{GOTO} M_j$
	WHILE $x_i \neq 0$ DO P END	$M_k : \mathbf{IF} x_i = c \mathbf{THEN GOTO} M_j$
		$M_k : \mathbf{HALT}$

Dabei darf jede Marke M_k nur einmal vorkommen

Klausurvorbereitung FSK-9-K Zeigen Sie, dass die Funktion *odd* ein primitiv rekursives Prädikat ist:

$$\text{odd}(x_1) = \begin{cases} 0 & \text{für } x_1 \text{ gerade} \\ 1 & \text{für } x_1 \text{ ungerade} \end{cases}$$

LÖSUNGSVORSCHLAG:

Hier genügt es, *odd* im primitiv rekursiven Schema anzugeben:

$$\text{odd}(x_1) = \begin{cases} 0 & \text{für } x_1 = 0 \\ \text{iszero}(\text{odd}(x_1 - 1)) & \text{sonst} \end{cases}$$

Die verwendeten Funktionen sind $g() = 0$ (konstante Funktion) und $h(y_1, y_2) = \text{iszero}(\pi_1^2(y_1, y_2))$ (Komposition).

Diese sind ebenfalls wieder in das primitiv rekursive Schema einzusetzen.