

## Prüfung Compiler

---

Datum	:	13.07.2017, 08:30 Uhr
Bearbeitungszeit	:	90 Minuten
Prüfer	:	Prof. Dr. Oliver Braun
Hilfsmittel	:	Keine
Erreichbare Punkte	:	90

---

Name: \_\_\_\_\_

Vorname: \_\_\_\_\_

Matrikelnummer: \_\_\_\_\_ Studiengruppe: \_\_\_\_\_

Hörsaal: \_\_\_\_\_ Platz Nr.: \_\_\_\_\_

Unterschrift: \_\_\_\_\_

Bitte kontrollieren Sie, ob Sie eine vollständige Angabe mit 5 Aufgaben auf 10 Seiten erhalten haben.

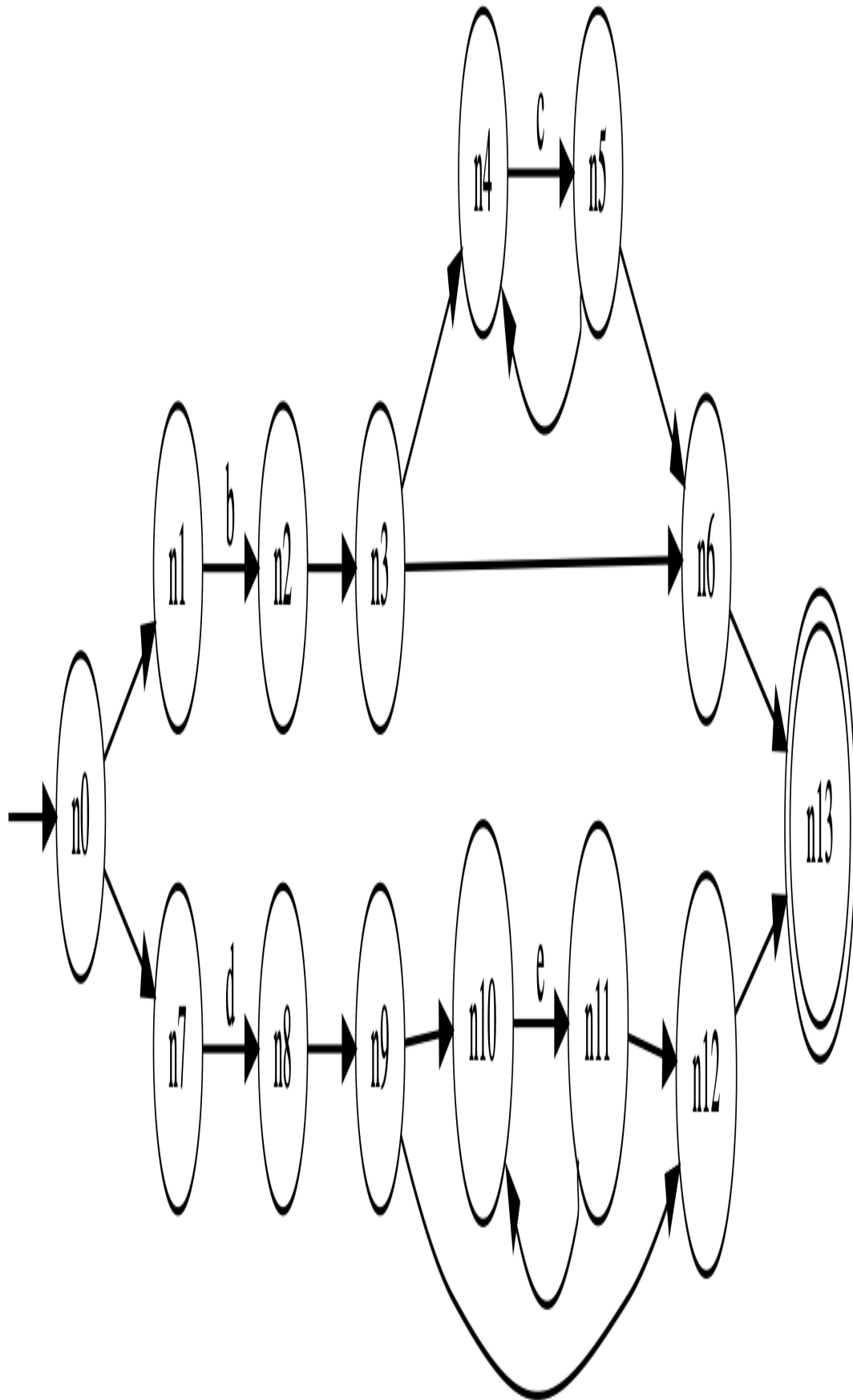
Aufgabe	1	2	3	4	5	Summe
max. Punkte	20	20	28	10	12	90

### Anmerkungen:

- Schreiben Sie die Lösungen in die dafür vorgesehenen Kästchen. Sollte Ihnen der Platz dabei nicht reichen, benutzen Sie die Rückseite **und vermerken Sie das im dazugehörigen Kästchen!**

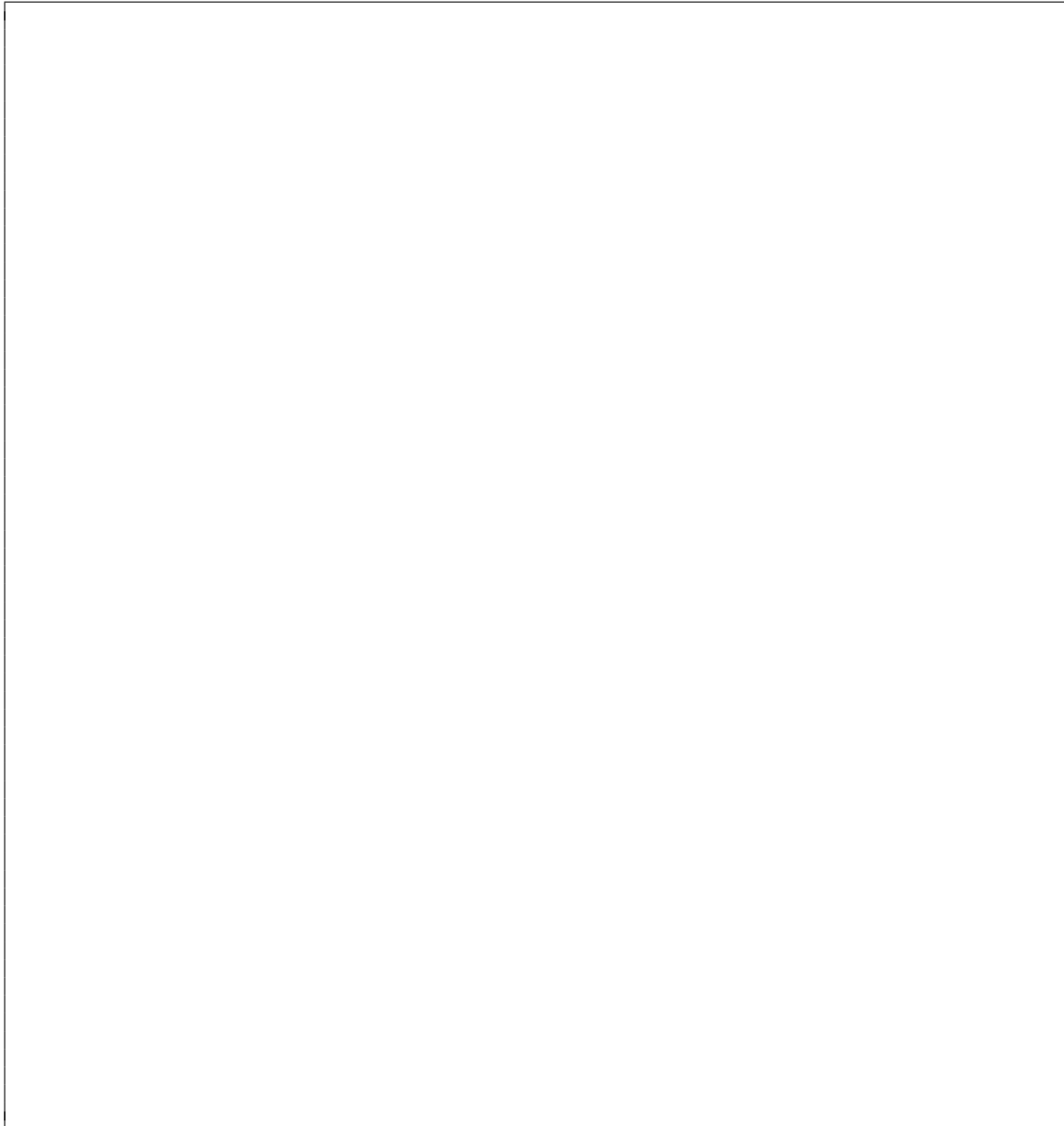
**Aufgabe 1 (20 Punkte)**

Gegeben sei folgender NFA:



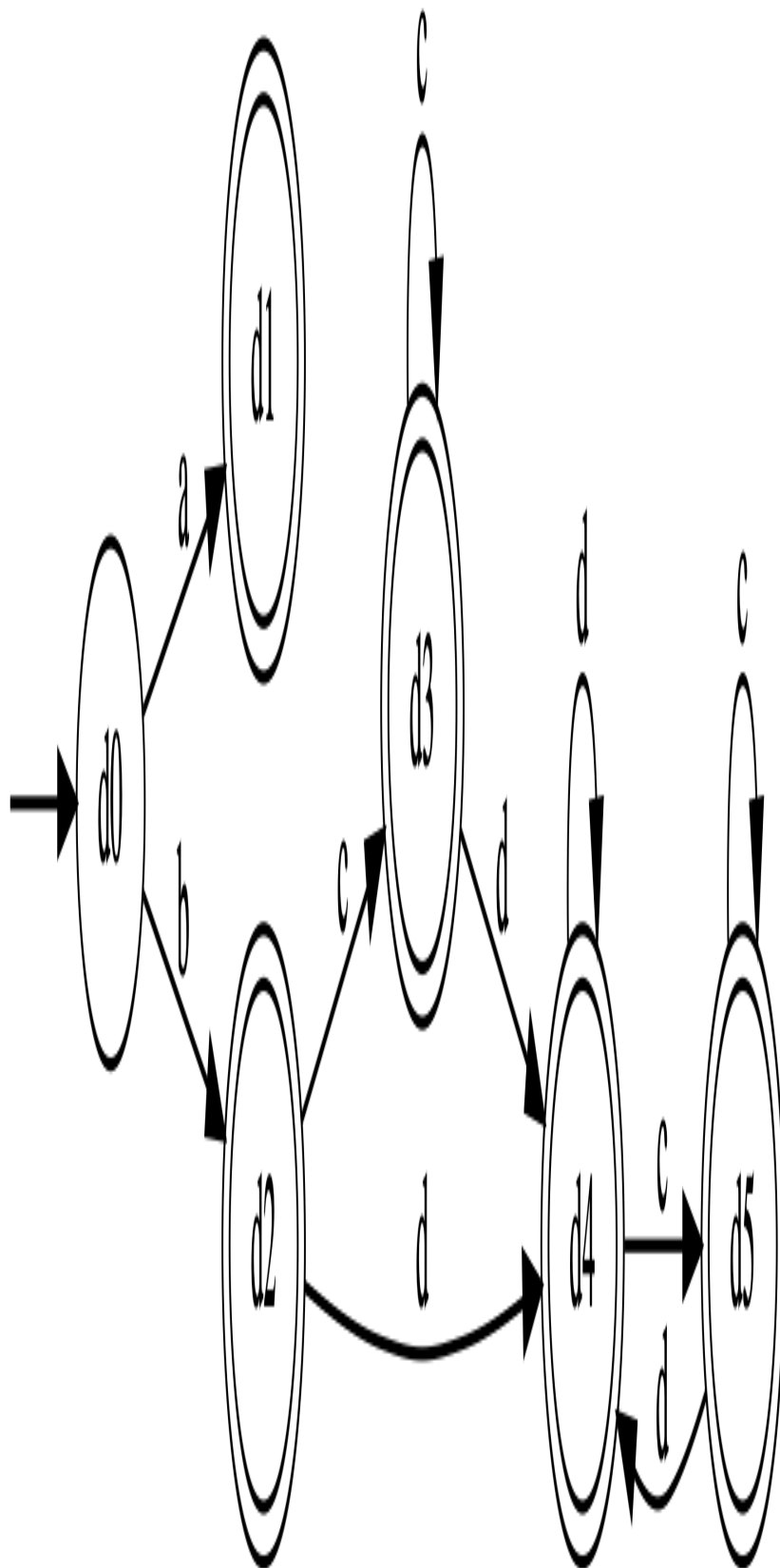
Erzeugen Sie mit Hilfe der, in der Vorlesung besprochenen, Teilmengenkonstruktion (*subset construction*) daraus einen DFA.

Geben Sie alle ermittelten Teilmengen **und** den resultierenden DFA an.

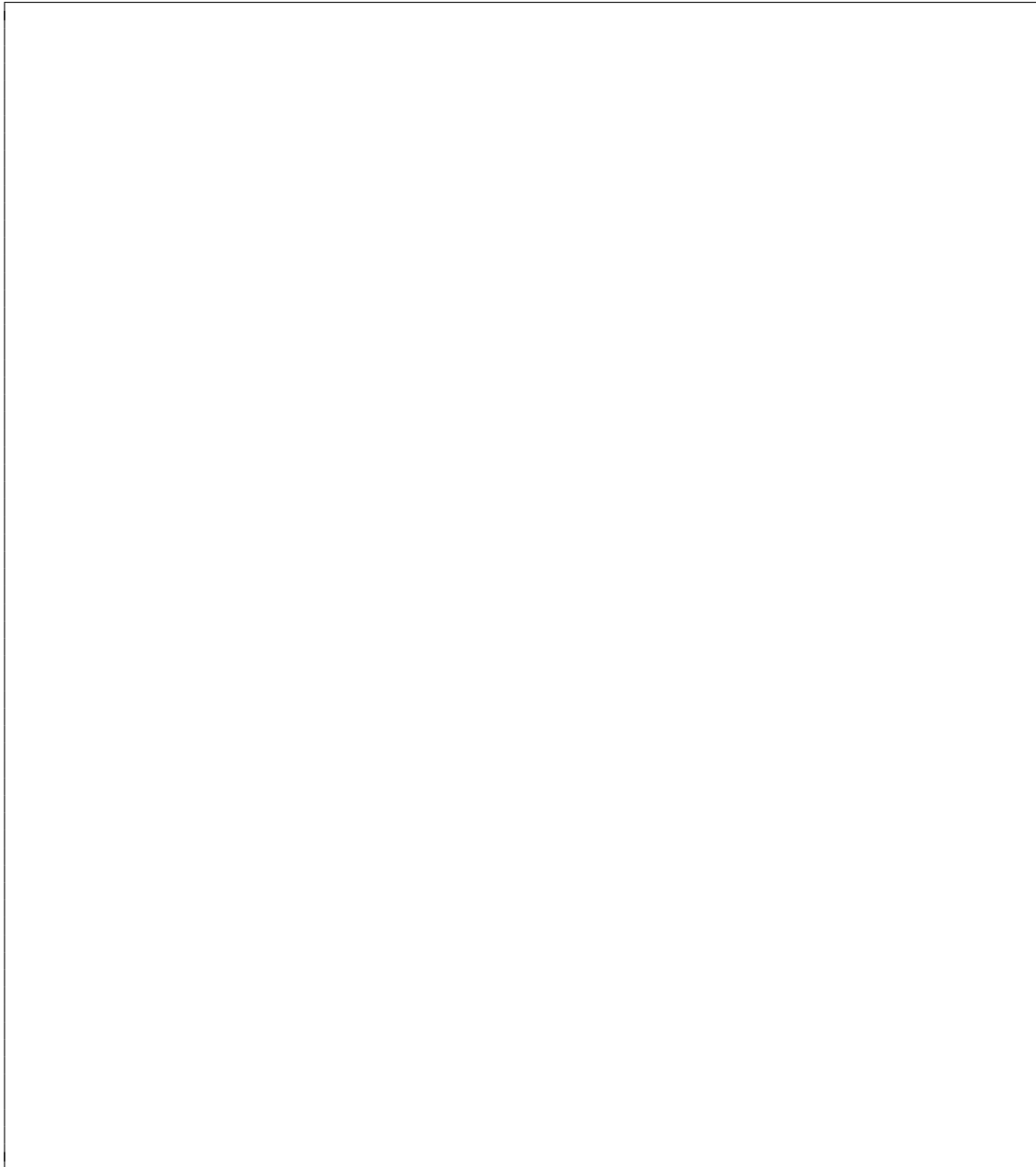


**Aufgabe 2 (20 Punkte)**

Gegeben sei folgender DFA:



Erzeugen Sie mit Hilfe von Hopcroft's Algorithmus daraus einen minimalen DFA.  
Geben Sie alle Partitionen **und** den resultierenden minimalen DFA an.



### Aufgabe 3 (28 Punkte)

Geben Sie für die folgenden regulären Ausdrücke jeweils einen **deterministischen endlichen Automaten**, der die gleichen Zeichenfolgen akzeptiert, als Zustandsübergangsdiagramm an. Sie brauchen die Zustände nicht benennen, es reicht wenn Sie kleine Kreise zeichnen. Sie brauchen den DFA nicht formal herleiten bzw. konstruieren.

(a)  $hal^*(o|a^*)$

(8)

(b)  $www^*w(w|ww^*)^*$

(8)

(c)  $(a^*|bc^*c)|ad(x^*|xd)$

(12)



**Aufgabe 4 (10 Punkte)**

Eliminieren Sie in der folgenden Grammatik die Linksrekursion.

---

0	<i>Goal</i>	$\mapsto$	<i>Expr</i>
1	<i>Expr</i>	$\mapsto$	<i>Expr</i> + <i>Term</i>
2			<i>Expr</i> - <i>Term</i>
3			<i>Term</i>
4	<i>Term</i>	$\mapsto$	<i>Term</i> * <i>Factor</i>
5			<i>Term</i> / <i>Factor</i>
6			<i>Factor</i>
7	<i>Factor</i>	$\mapsto$	( <i>Expr</i> )
8			num
9			name

---

Sie müssen nur die geänderten/neuen Regeln angeben.

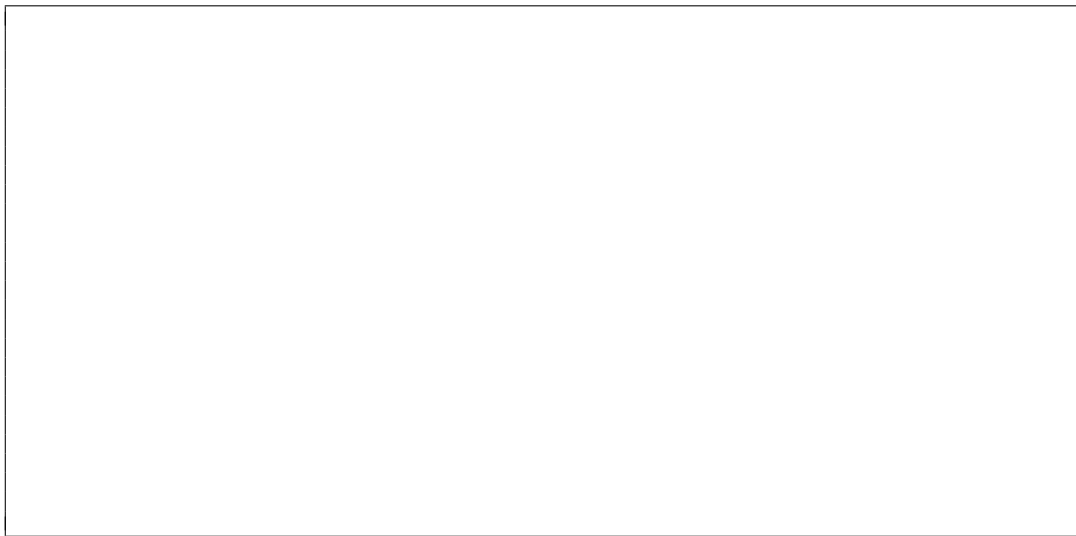
### Aufgabe 5 (12 Punkte)

Gegeben sei folgender Java-ähnlicher Quellcode:

```
1  int x = 12;
2  int y = 7;
3  while (y<12) {
4      int a = 10;
5      while (a>0) a--;
6      if (y!=x)
7          a = x-- + --y;
8      else
9          a = 5
10     y--;
11 }
```

(a) Geben Sie den Kontrollflußgraph (*control-flow graph*) für obigen Code an.

(6)



(b) Geben Sie den Abhängigkeitsgraph (*dependence graph*) für obigen Code an.

(6)

