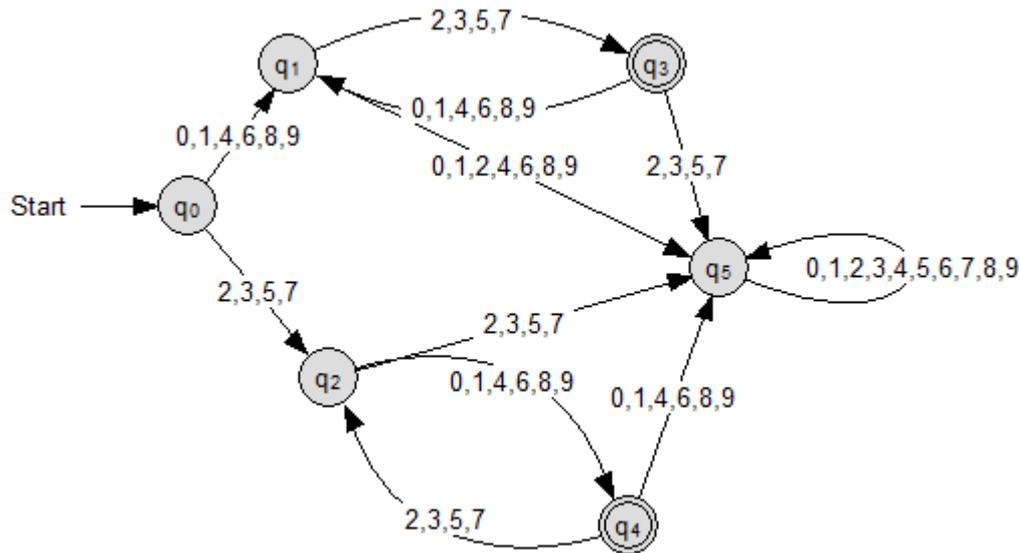


Informatik Abitur Bayern 2011 / III - Beispiellösung					Autor Rank	
1a)	Befehl	ac	program counter	101	102	3
	(Anfangszustand)	0	1	5	7	
	LOAD 102	7	2	5	7	
	SUB 101	2	3	5	7	
	JGE 6	2	6	5	7	
	END	2	7	5	7	
b)	<p>i) Paar (a q) nach jedem Durchlauf: (15 0) → (10 1) → (5 2) → (0 3)</p> <p>Am Ende: q = 3; und r = a = 0;</p> <p>ii) (13 0) → (8 1) → (3 2)</p> <p>Am Ende: q = 2; und r = a = 3;</p> <p>Es wird der ganzzahlige Quotient q und der Teilungsrest r bei der Division von a durch b berechnet.</p>					3
c)	<pre>wdh: LOAD 101 SUB 102 -- a - b JLT endwdh -- falls a < b STORE 101 -- a = a - b DLOAD 1 ADD 103 STORE 103 -- q = q + 1 JUMP wdh endwdh: LOAD 101 STORE 104 -- r = a END</pre>					7
2	<p>Zeit = Std ' : ' Min ' : ' Sek;</p> <p>Std = ('0' 1) Ziffer ' 2 ' ('0' '1' '2' '4');</p> <p>Min = ('0' '1' '2' '4' '5')Ziffer;</p> <p>Sek = Min [',' Ziffer [Ziffer]] ;</p> <p>10erZiffer = '0' '1' '2' '4' '5' '6' '7' '8' '9';</p>					8

3a)

6



Fehlerzustand q_5 ist optional.

b)

3

Ziffernfolge: 4567

$\rightarrow q_0 \xrightarrow{4} q_1 \xrightarrow{5} q_3 \xrightarrow{6} q_1 \xrightarrow{7} q_3$

Da der Zustand q_3 ein Endzustand ist, akzeptiert der Automat die Ziffernfolge.

Ziffernfolge: 123456

$\rightarrow q_0 \xrightarrow{1} q_1 \xrightarrow{2} q_3 \xrightarrow{3} q_5 \xrightarrow{4} q_5 \xrightarrow{5} q_5 \xrightarrow{6} q_5$

Die Ziffernfolge wird nicht akzeptiert, da der Automat in einem nicht akzeptierenden Zustand hält.

c)

10

Im folgenden ist die komplette Lösung dargestellt. Laut Aufgabenstellung genügt es auch nur zwei der case-Fälle der Methode `zustandWechseln` zu implementieren. Es wird auf die Implementierung eines Default-Falles verzichtet, da davon ausgegangen wird, dass nur Tasten auf dem Ziffernblock aktiviert sind.

```

public boolean folgePrüfen(ZEICHENKETTE wort) {
    //für jede Ziffer wird die Methode zustandWechseln aufgerufen.
    for (int i = 0; i < wort.laenge(); i++) {
        zustandWechseln(wort.zeichenAnPosition(i));
    }
}
  
```

```

// wenn der Zustand 3 oder 4 als Endzustand erreicht ist, wird
true zurück gegeben
return (zustand == 3 || zustand == 4);
}

public void zustandWechseln(ZEICHEN zeichen){
//je nach Zustand wird der entsprechende Fall ausgeführt, indem
je nach Primzahl 'P' oder keine Primzahl 'N' unterschieden wird
um in den entsprechenden Folgezustand zu wechseln
switch (zustand) {
    case 0: {
        if(zeichen=='N') {zustand = 1;} else {zustand = 2;}
    } break;
    case 1: {
        if(zeichen=='N') {zustand = 5;} else {zustand = 3;}

    } break;
    case 2: {
        if(zeichen=='N') {zustand = 4;} else {zustand = 5;}

    } break;
    case 3: {
        if(zeichen=='N') {zustand = 1;} else {zustand = 3;}

    } break;
    case 4: {
        if(zeichen=='N') {zustand = 5;} else {zustand = 2;}
    } break;
//case 5 unnötig, da keine Zustandsänderung aus dem Fangzustand
heraus geschieht.
    }
}
}

```