Test und Verlässlichkeit
Grosse Übung 1

Prof. G. Kemnitz

Institut für Informatik, Technische Universität Clausthal

6. Mai 2015

C-typischer Multiplikationsfehler

Aufgabe F1-1.2: C-typischer Multiplikationsfehler

Eine Service-Leistung sei definiert durch:

- Eingabeformat: zwei Variablen a und b, 8-Bit vorzeichenfrei
- Ausgabeformat: Rückgabewert 16-Bit vorzeichenfrei
- Sollfunktion: Rückgabe des Produkts a*b
- Implementierung als C-Funktion:

```
uint16_t umult16(uint8_t a, uint8_t b){
 return a*b;
}
```

1. Kleinster und größter darstellbarer Ein- und Ausgabewerte?
2. Für welche Bedatungen von a und b unterscheidet sich der Ist- vom Soll-Wert der Ausgabe[1]?
3. Wie ist die Ist-Funktion zu verändern, dass für alle Eingabewerte das korrekte Ergebnis berechnet wird?

[1] In C hat ein Produkt den Typ des Operanden mit dem größten Wertebereichs. Typenumwandlung der Zuweisung erst nach Produktbildung.

1. C-typischer Multiplikationsfehler

1. Kleinster und größter darstellbarer Ein- und Ausgabewerte?
   - Wertebereich von uint8_t (unsigned char): 0 bis 255
   - Wertebereich von uint16_t (unsigned short): 0 bis $2^{16} - 1$
   - Ein 8 Bit $\times$ 8 Bit Produkt hat »oft« in C auch nur 8 Bit.
2. Für welche Bedatungen von a und b unterscheidet sich der Ist- vom Soll-Wert der Ausgabe?
   - Für $a \cdot b > 255$.
3. Fehlerbeseitigung?
   - Typcast auf 16 Bit für mindestens einen Summanden:

```
uint16_t umult16(uint8_t a, uint8_t b){
 return ((uint16_t)a)*b;
}
```

Typ. Fehler einer Gleitkommadivision

Aufgabe F1-1.3: Fehler einer Gleitkommadivision

Eine Service-Leistung sei definiert durch:

- Ein- und Ausgabegabeformat: 32-Bit Gleitkommaformat IEEE 754 »single«
- Soll-Funktion: Rückgabe von y=sin(x)/x mit maximaler Soll-/Ist-Abweichung:

$$\frac{|y_{\text{Soll}} - y_{\text{Ist}}|}{y_{\text{Ist}}} < 0.01\%$$

- Implementierung als C-Funktion:

```c
#include <math.h>
float sinc(float x){
  return sin(x)/x;
}
```

2. Typ. Fehler einer Gleitkommadivision

1. Beschreiben Sie den Aufbau das Gleitkommaformats IEEE 754 »single«[2].
2. Wie wird der Eingabewert -5.0 dargestellt?
3. Für welchen Eingabebereich weicht das Ist-Ergebnis vom Soll-Ergebnis ab.
4. Verbessern Sie die Implementierung, so dass sie auch für den im Aufgabenteil zuvor bestimmten Wertebereich der Eingabe korrekte Ergebnisse liefert.

[2] Die benötigten Informationen finden unter dem Suchbegriff »IEEE Gleitkommaformat« im Internet.

Gleitkommaformats IEEE 754 »single«

IEEE 754 »single« ist ein 32-Bit-Format:

- Bit 31: Vorzeichenbit s
- Bit 30..23: Charakteristik c(Kommaverschiebung)
- Bit22..0: Mantisse M: Wertebereich

Wert für $0 < c < 255$ (normierte Darstellung[3]):

$$Z = (-1)^s \cdot (1, M_{-1} \ldots M_{-m}) \cdot 2^{c-127}$$

Wert für $c = 0$ (denormiert[4]):

$$Z = (-1)^s \cdot (M_0, M_{-1} \ldots M_{-m}) \cdot 2^{-127}$$

[3] In der normierten Darstellung ist die Mantisse M eine vorzeichenfreie Zahl mit einem Vorkommabit gleich eins, das nicht mit gespeichert wird.

[4] In der denormierten Darstellung kann das Vorkommabit auch null sein und wird mit gespeichert.

Sonderwerte $c = 255$:

$$Z = \begin{cases} \infty & \text{für } s = 0 \text{ und } m = 0 \\ -\infty & \text{für } s = 1 \text{ und } m = 0 \\ \text{nan} & \text{für } m \neq 0 \end{cases}$$

(nan, not a number – ungültig; $\pm\infty$ – positiver/negativer Wertebereichsüberlauf)

- Darstellung von »-5«:

$$-5 = (-1)^1 \cdot 1.01 \cdot 2^{129-127}$$

$(s = 1; c = 0x81; M = 010...0)$

Bit: 31	30 ... 23	22 ... 0
1	1 0 0 0 0 0 0 1	0 1 0 0 0 0 0 0 0 0 ... 0
Vorteichen	Charakteristik	Mantisse ohne Vorkommanull

- Fehler in

```
#include <math.h>
float sinc(float x){
  return sin(x)/x;
}
```

  Für $x = 0$ liefert die Division $0/0 \rightarrow$nan, Sollwert ist aber 1.

- Mögliche Fehlerbeseitigung:

```
float sinc(float x b){
  if (x==0) return 1.0;
  return sin(x)/x;
}
```

Initialisierungsfehler

Aufgabe F1-1.4: Initialisierungsfehler

Das nachfolgende Unterprogramm soll für das mit einem Zeiger auf den Anfang und der Länge übergebene Feld den kleinsten Wert zurückgeben und hat ein unbeständiges Fehlverhalten.

```
int16_t Feld[]= {231, -13, ...}; // Beispiel für ein Feld
...
int16_t kleinsterWert(int16_t *Feld, uint16_t len){
  int16_t tmp, *ptr;
  for (ptr=Feld; ptr < Feld+len; ptr++){
    if (*ptr<tmp) tmp = *ptr;
  }
 }
```

1. Mit welchen Eingaben und Zusatzbedingungen ist der Fehler nachweisbar?
2. Ändern Sie das Programm so, dass es korrekt funktioniert.

- Es wird nicht der kleinste Wert des Eingabefeldest, sondern des Eingabefeldes und des zufälligen Initialwertes von »tmp«, den die letzte lokale Variable auf dieser Adresse hatte, gebildet.
- Der Fehlernachweis ist um so wahrscheinlicher, je größer die Werte im übergebenen Feld sind.
- Korrigiertes Programm:

```
int16_t kleinsterWert(int16_t *Feld, uint16_t len){
  assert (len>0) ....; // Bedingung zul. Eingabe
  int16_t tmp=Feld[0], *ptr;
  for (ptr=Feld+1; ptr <Feld+len; ptr++){
    if (*ptr<tmp) tmp = *ptr;
  }
 }
```

Fehler in kombinatorischer Schaltung

Aufgabe F1-1.5: Fehler in komb. Schaltung

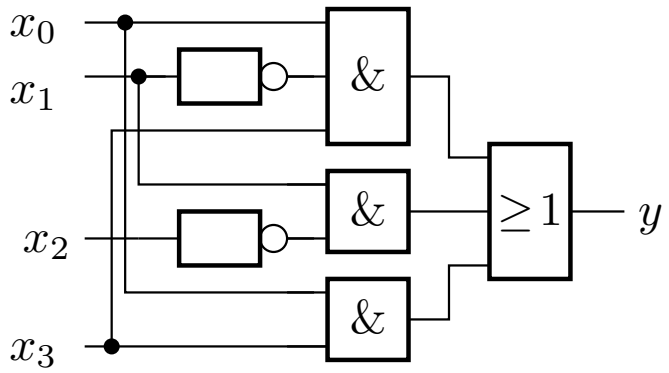
Eine kombinatorische Schaltung mit der Soll-Funktion entsprechend der nachfolgenden Wertetabelle ist durch die Schaltung daneben realisiert.

Soll-Funktion

x_3	x_2	x_1	x_0	y
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0

x_3	x_2	x_1	x_0	y
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0

Realisierung

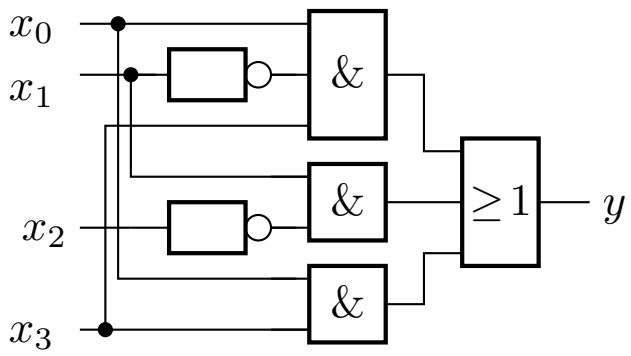


1. Stellen Sie die Wertetabelle der Realisierung auf. Für welche Eingaben weicht die Ausgabe vom Soll-Wert ab.
2. Verbessern Sie die Realisierung so, dass Sie für alle Eingaben richtige Ergebnisse liefert.

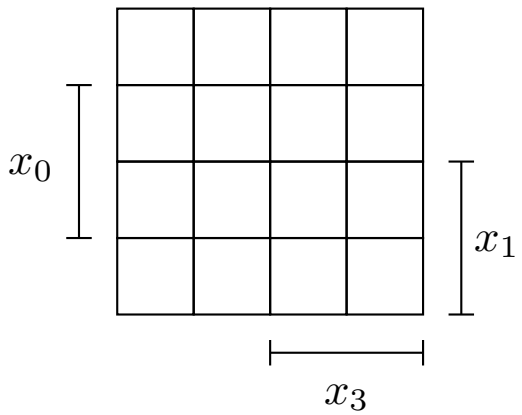
x_3	x_2	x_1	x_0	y
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0

x_3	x_2	x_1	x_0	y
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0

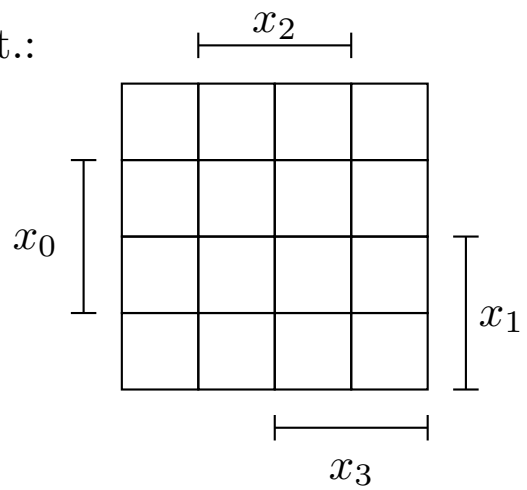
Realisierung



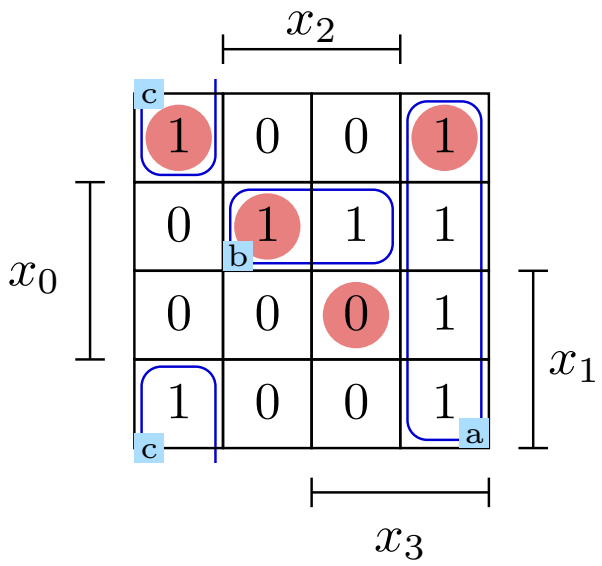
Sollfkt.:



Istfkt.:



- Soll- und Ist-Funktion in KV-Diagramm übernehmen.
- KV-Diagramm Istfkt.: Abweichungen kennzeichnen.
- KV-Diagramm Sollfkt.: richtige Schaltung konstruieren.



Falsche Ausgabe

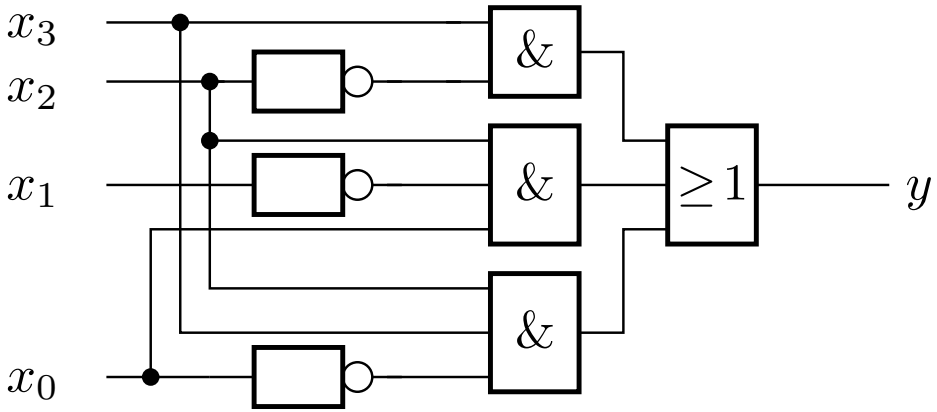
korrigierte Gleichungen:

a: $x_3\bar{x}_2$

b: $x_2\bar{x}_1x_0$

c: $x_3x_2\bar{x}_0$

$y = x_3\bar{x}_2 \vee x_2\bar{x}_1x_0 \vee x_3x_2\bar{x}_0$



Wahrscheinlichkeiten von Würfelexperimenten

Aufgabe F1-2.1: Würfelexperimenten

X und Y seien die zufälligen Augenzahlen bei der Durchführung des Versuchs »Würfeln mit zwei Würfeln«. Berechnen Sie die Wahrscheinlichkeiten folgender Ereignisse:

1. $X + Y > 8$
2. $X > Y$
3. $(X = 5) \wedge (Y < 5)$
4. $X \cdot Y$ ist durch drei teilbar.

Geben Sie jeweils die Anzahl der möglichen Ereignisse an und zählen Sie die günstigen Ereignisse auf.

- $2\times$ würfeln hat die 36 mögliche Ergebnisse: $(1,1)$, ..., $(6,6)$.

1. Günstige Ergebnisse $X + Y > 8$: $(3,6), \ldots$
2. Günstige Ergebnisse $X > Y$: $(2,1), \ldots$
3. Günstige Ergebnisse $(X = 5) \wedge (Y < 5)$: $(5,1)$, ...
4. Günstige Ergebnisse $X \cdot Y$ ist durch drei teilbar: $(3,1)$, ...

Verkettete Würfelereignisse

Aufgabe F1-2.2: Verkettete Würfelereignisse

- Welche möglichen Ergebnisse hat das Zufallsexperiment »auswürfeln einer Zahl, bei einer Sechs darf ein zweites Mal gewürfelt werden«?
- Mit welcher Wahrscheinlichkeit tritt jedes der möglichen Ergebnisse ein?

Lösung:

- einmal würfeln: 1,2,...,5 mit $p = \frac{1}{6}$
- eine sechs: 7,8,...,11 mit $p = \frac{1}{6^2}$
- zwei sechsen: 13,14,...17 mit $p = \frac{1}{6^3}$
- ...
- die Werte 6, 12, 18, ... können nicht auftreten.

Fehlfunktionen und Fehlernachweis

Aufgabe F1-2.3: Fehlfunktionen und Fehlernachweis

Ein System habe vier unabhängig voneinander nachweisbare Fehler mit den Nachweiswahrscheinlichkeiten je Service-Aufruf von $p_1 = 10\%$, $p_2 = 20\%$, $p_3 = 5\%$ und $p_4 = 1\%$.

1. Mit welcher Wahrscheinlichkeit versagt eine einzelne Service-Anforderung?
2. Wie hoch ist die Wahrscheinlichkeit, dass zehn aufeinanderfolgende Service-Anforderungen korrekt ausgeführt werden?
3. Wie groß ist die Wahrscheinlichkeit für jeden der vier Fehler, dass er bei einem der zehn aufeinanderfolgenden Service-Aufrufe nachgewiesen wird (mindestens ein Versagen verursacht)?

- Mit welcher Wahrscheinlichkeit versagt eine einzelne Service-Anforderung?

  Versagen tritt ein, wenn ein Fehler oder nicht kein Fehler nachweisbar ist:

$$
\begin{aligned}
V &= F_1 \vee F_2 \vee F_3 \vee F_4 \\
V &= \overline{\bar{F}_1 \bar{F}_2 \bar{F}_3 \bar{F}_4} \\
P(V) &= 1 - (1 - P(F_1)) \cdot (1 - P(F_2)) \cdot (1 - P(F_3)) \cdot (1 - P(F_4)) \\
&= 1 - 0{,}9 \cdot 0{,}8 \cdot 0{,}95 \cdot 0{,}99 = 23{,}3\%
\end{aligned}
$$

- Wie hoch ist die Wahrscheinlichkeit, dass zehn aufeinanderfolgende Service-Anforderungen korrekt ausgeführt werden?

  Wahrscheinlichkeit, dass in keine der 10 Anforderungen versagt:

$$P(V10) = (1 - P(V))^{10} = (1 - 23{,}3\%)^{10} = 2\%$$

- Wie groß ist die Wahrscheinlichkeit für jeden der vier Fehler, dass er bei einem der zehn aufeinanderfolgenden Service-Aufrufe nachgewiesen wird (mindestens ein Versagen verursacht).
  Jeweils die Wahrscheinlichkeit, dass nicht keine der Anforderungen den Fehler nachweisen:

$$
\begin{aligned}
p_1\,(n=10) &= 1-(1-10\%)^{10} = 65\% \\
p_2\,(n=10) &= 1-(1-20\%)^{10} = 89\% \\
p_3\,(n=10) &= 1-(1-\ \ 5\%)^{10} = 40\% \\
p_4\,(n=10) &= 1-(1-\ \ 1\%)^{10} = 9{,}6\%
\end{aligned}
$$

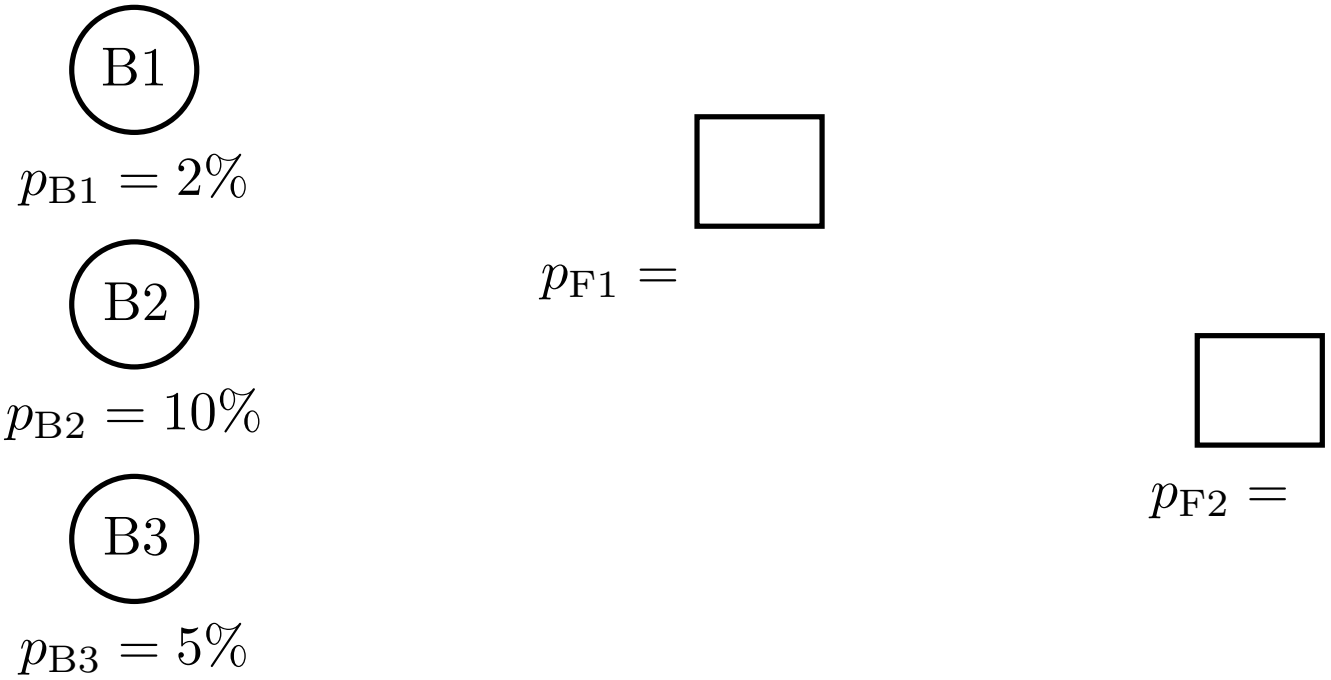
Fehlerbaumanalyse

Aufgabe F1-2.5: Fehlerbaumanalyse

1. Entwickeln Sie den Fehlerbaum für folgenden Zusammenhang:
   - Ereignis F_1 tritt ein, wenn entweder B_1 und nicht B_2 oder nicht B_1 und B_2 eintritt.
   - Das Ereignis F_2 tritt nur ein, wenn F_1 und B_3 eintreten.

2. Berechnen Sie die Wahrscheinlichkeit für F_1 und F_2 für den Fall, dass die Wahrscheinlichkeiten der Basisereignisse $p_{\mathrm{B1}} = 2\%$, $p_{\mathrm{B2}} = 10\%$ und $p_{\mathrm{B3}} = 5\%$ betragen.

- Ereignis F_1 tritt ein, wenn entweder B_1 und nicht B_2 oder nicht B_1 und B_2 eintritt.
- Das Ereignis F_2 tritt nur ein, wenn F_1 und B_3 eintreten.



$$
\begin{aligned}
P\left(\mathrm{B1} \wedge \overline{\mathrm{B2}}\right) &= p_{\mathrm{B1}} \cdot (1 - p_{\mathrm{B2}}) = 2\% \cdot 90\% = 1{,}8\% \\
P\left(\mathrm{B2} \wedge \overline{\mathrm{B1}}\right) &= p_{\mathrm{B2}} \cdot (1 - p_{\mathrm{B1}}) = 10\% \cdot 98\% = 9{,}8\% \\
P\left(\mathrm{F1}\right) &= P\left(\mathrm{B1} \wedge \overline{\mathrm{B2}}\right) + P\left(\mathrm{B2} \wedge \overline{\mathrm{B1}}\right)^{*} = 1{,}8\% + 9{,}8\% = 11{,}6\% \\
P\left(\mathrm{F2}\right) &= P\left(\mathrm{F1}\right) = 11{,}6\% \cdot 5\% = 0{,}58\%
\end{aligned}
$$

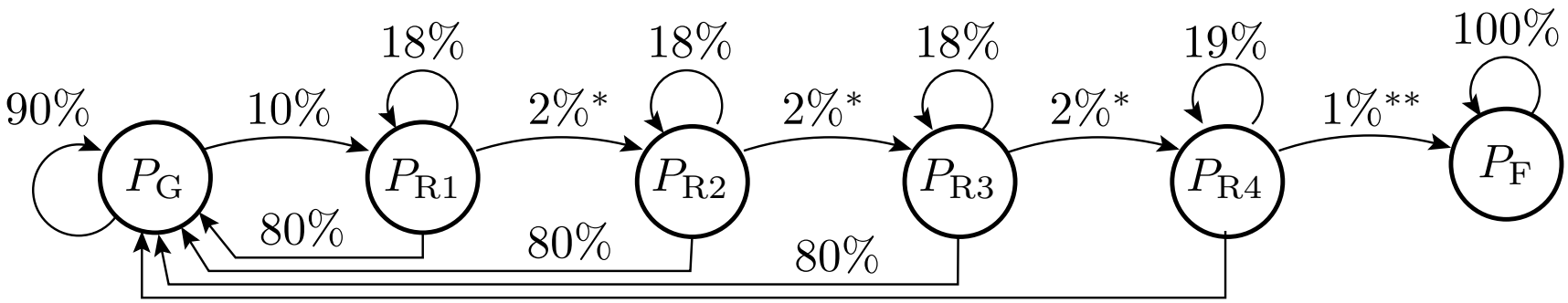
(* Die Bedingungen $\mathrm{B1} \wedge \overline{\mathrm{B2}}$ und $\mathrm{B2} \wedge \overline{\mathrm{B1}}$ schließen sich aus.)

Risikoanalyse

Aufgabe F1-2.7: Risikoanalyse

Eine schwerwiege Fehlfunktion bei einer Maschine kann nur auftreten, wenn sie vom Normalzustand Z_0 nacheinander in höhere Riskozustände R_1 bis R_4 übergeht. Das Bedienpersonal erkennt erhöhte Risikozustände mit einer Wahrscheinlichkeit 80% und initialisiert das System dann neu (Rückkehr in den Grundzustand G). Die Wahrscheinlichkeit für den Übergang von einem in den nächsten Risikozustand betrage in jedem Zeitschritt, wenn nicht neuinitialisiert wird, 10%. In Risikozustand R_4 tritt ohne rechtzeitige Neuinitialisierung mit 5% die schwerwiegende Fehlersituation F ein.

1. Beschreiben Sie den Sachverhalt mit einer Markov-Kette.
2. Simulation der Markov-Kette mit Matlab oder Octave für 10 Schritte.
3. Zusammenhangs $P(F)$ und Zeitschrittanzahl n für weniger als 10^6 Zeitschritte?



*10%, wenn keine Neuinitilaisierung ** 5%, wenn keine Neuinitilaisierung

```
PN = 100; PR1 = 0; PR2=0; PR3=0; PR4=0; PF=0;
fprintf('  n|  P(N)|  P(R1)| P(R2)| P(R3)| P(R4)  |  P(F)\n');
for n = 1:10
 PN  = PN *0.9 + PR1*0.8 + PR2*0.8 + PR3*0.8 + PR4*0.8;
 PR1 = PN *0.10 + PR1*0.18;
 PR2 = PR1*0.02 + PR2*0.18;
 PR3 = PR2*0.02 + PR3*0.18;
 PR4 = PR3*0.02 + PR4*0.19;
 PF  = PR4*0.01 + PF;
 fprintf('%3i|%6.3f| %6.3f|%6.3f|%6.3f|%8.6f|%8.6f\n',
            n, PN, PR1, PR2, PR3, PR4, PF);
end;
```

n	P(N)	P(R1)	P(R2)	P(R3)	P(R4)	P(F)
1	90.000	9.000	0.180	0.004	0.000072	0.000001
2	88.347	10.455	0.241	0.005	0.000123	0.000002
3	88.074	10.689	0.257	0.006	0.000146	0.000003
4	88.029	10.727	0.261	0.006	0.000154	0.000005
5	88.021	10.733	0.262	0.006	0.000157	0.000007
6	88.020	10.734	0.262	0.006	0.000157	0.000008
7	88.020	10.734	0.262	0.006	0.000158	0.000010
8	88.020	10.734	0.262	0.006	0.000158	0.000011
9	88.020	10.734	0.262	0.006	0.000158	0.000013
10	88.020	10.734	0.262	0.006	0.000158	0.000014

Für P(N) bis P(R4) stellt sich nach 5 Schritten ein stationärer Zustand ein. Für $P(F) \ll 100\%$ gilt etwa:

$$P(F) = 1{,}58 \cdot 5\% \cdot n$$

Nach 1 Millionen Zeitschritten $P(F) \approx 1{,}5\%$:

n	P(N)	P(F1)	P(F2)	P(F3)	P(F4)	PF
1000000	86.491	10.548	0.257	0.006	0.000155	1.562945