Test und Verlässlichkeit Grosse Übung zu Foliensatz 5: Überwachung, Fehlerbehandlung und Fehlertoleranz

Prof. G. Kemnitz

1. Juli 2020

Contents

2 Informationsredundanz

2.1 Fehlererk. Codes

Aufgabe 5.1: Arithmetischer Code

a) Bilden Sie für den Bitvektor

$$x = 110010001000011101_2$$

das fehlererkennende Codewort durch Multiplikation seines Wertes als vorzeichenfreie ganze Binärzahl mit der Primzahl $c = 10313$ (Bestimmung des Dezimalwerts, Multiplikation und Konvertierung des Produkts in einen Binärvektor).

b) Mit welcher Wahrscheinlichkeit werden mit dem gewählten fehlererkennenden Code Datenverfälschungen des codierten Bitvektors $s = c \cdot x$ erkannt?

c) Werden mit dem gewählten Code Verfälschung von s erkannt, die die Bitstellen 3 und 14 invertieren? Hinweis: Eine Verfälschung von s ist am Divisionsrest zu erkennen, wenn die Abweichung zum Sollwert $\Delta s = s - s_{\text{soll}}$ kein Vielefachses des Multiplikators c ist.

Eingabewert hexadezimal: 11.0010.0010.0001.1101 = 0x3221D

- Mit Octave (Matlab) Produkt als hexadezimal:

```
>> printf('CW=0x%x\n',0x3221D*10313)
CW=0x7e394245
```

binär: 0b111.1110.0011.1001.0100.0010.0100.0101

b) Erkennungswahrscheinlichkeit:

$$p_{\mathrm{E}} \approx 1 - \frac{1}{10313} = 99{,}990\%$$

c) Keine Maskierung, wenn Bit 3 und 14 invertiert ist:

$$\mathrm{Rest}\left(\frac{0\mathrm{b}100.0000.0000.1000}{10313}\right) \neq 0\surd$$

Für Differenzen ungleich null, die kleiner als der Quotient sind, immer erfüllt.

2.3 Prüfkennzeichen

Aufgabe 5.2: Prüfsummen
Bilden Sie für die Bytefolge

`0x13, 0xF2, 0x33, 0xE6`

die Prüfsumme:

a) durch byteweises Aufsummieren unter Vernachlässigung der Überträge.

b) durch bitweise EXOR-Verknüpfung der Bytes.

c) Welche der beiden Prüfsummen erkennt, dass die nachfolgenden Datenfolgen verfälscht sind? c) Welche der b

```
F1: 0x13, 0x33, 0xF2, 0xE6
F2: 0x13, 0xF2, 0x37, 0xE6
F3: 0x13, 0xF1, 0x90, 0x56
```

Wert unverf.	(Teil-) Prüfsum.	binär
0x13		
0xF2		
0x33		
0xE6		
	EXOR:	

Wert	(Teil-) Prüfsum.	binär
0x13	0x13	0001 0011
0xF2	0x05	1111 0010
0x33	0x38	0011 0011
0xE6	0x1E	1110 0110
	EXOR:	0011 0100

Wert unverf.	(Teil-) Prüfsum.	binär
0x13		
0xF2		
0x33		
0xE6		
	EXOR:	

Wert F1	(Teil-) Prüfsum.	binär
0x13		
0x33		
0xF2		
0xE6		
	EXOR:	

Wert F2	(Teil-) Prüfsum.	binär
0x13		
0xF2		
0x37		
0xE6		
	EXOR:	

Wert F3	(Teil-) Prüfsum.	binär
0x13		
0xF1		
0x90		
0x56		
	EXOR:	

Wert	(Teil-) Prüfsum.	binär
0x13	0x13	0001 0011
0xF2	0x05	1111 0010
0x33	0x38	0011 0011
0xE6	0x1E	1110 0110
	EXOR:	0011 0100

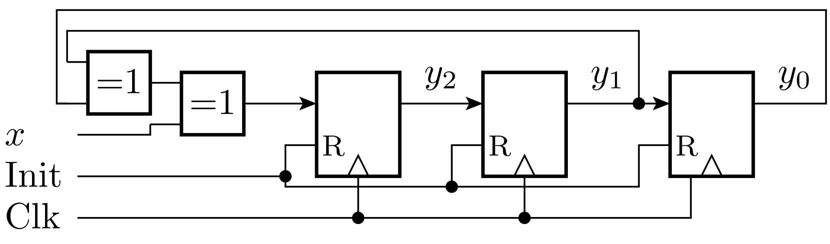
Wert	(Teil-) Prüfsum.	binär
0x13	0x13	0001 0011
0x33	0x46	0011 0011
0xF2	0x38	1111 0010
0xE6	0x1E	1110 0110
	EXOR:	0011 0100

Wert	(Teil-) Prüfsum.	binär
0x13	0x13	0001 0011
0xF2	0x05	1111 0010
0x37	0x3C	0011 0111
0xE6	0x22	1110 0110
	EXOR:	0011 0000

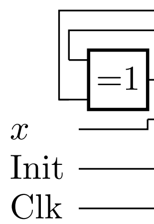
Wert	(Teil-) Prüfsum.	binär
0x13	0x13	0001 0011
0xF1	0x04	1111 0001
0x90	0x94	1001 0000
0x46	0xDA	0100 0110
	EXOR:	0011 0100

Aufgabe 5.3: Prüfkennzeichen mit LFSR

Gegeben ist folgendes linear rückgekoppelte Schieberegister:



	x	y_2	y_1	y_0
0	1	0	0	0
1	0			
2	1			
3	1			
4	0			
5	0			
6	1			
7	1			
8	0			
9	1			
10	0			
11	0			
12	1			
13	0			
14	1			
15	0			
PKZ:				



a) Auf welches Prüfkennzeichen $\mathbf{y} = y_2y_1y_0$ wird die Datenfolge 1011 0011 0100 1010 beginnend mit dem linken Bit und Startwert 000 abgebildet? Füllen Sie dazu die Tabelle in der Abbildung aus.

b) Wie hoch ist Fehlererkennungswahrscheinlichkeit?

$$p_{\mathrm{E}} \approx 1 - 2^{-3} = 87{,}5\%$$

2.5 Hamming-Codes

Aufgabe 5.4: Kreuzparität

a) Ergänzen Sie Bitwerte für die Längs- und Querparität.

b) Woran ist eine Invertierung des rot unterlegten Bits zu erkennen?

	Längsparität
1011001001101000	
1100001110010011	
0110010010101101	
1000100001100101	
1101001011010011	
1101000010011110	
1010011000010101	
1011010010100110	
Querparität:	

	Längsparität
1011001001101000	1
1100001110010011	0
0110010010101101	0
1000100001100101	0
1101001011010011	1
1101000010011110	0
1010011000010101	1
1011010010100110	0
Querparität: 1000110111001111	

Die Invertierung des rot unterlegten Bits ist an einem Paritätsfehler in Zeile 6 und in Spalte 7 zu erkennen.

Aufgabe 5.5: (8,12)-Hamming-Code

b_{12}	b_{11}	b_{10}	b_9	b_8	b_7	b_6	b_5	b_4	b_3	b_2	b_1
x_7	x_6	x_5	x_4	q_3	x_3	x_2	x_1	q_2	x_0	q_1	q_0

$$\begin{aligned} q_0 &= x_0 \oplus x_1 \oplus x_3 \oplus x_4 \oplus x_6 \\ q_1 &= x_0 \oplus x_2 \oplus x_3 \oplus x_5 \oplus x_6 \\ q_2 &= x_1 \oplus x_2 \oplus x_3 \oplus x_7 \\ q_3 &= x_4 \oplus x_5 \oplus x_6 \oplus x_7 \end{aligned}$$

a) Bilden Sie die Codeworte für die darzustellenden Werte: $w_1 = $ 0x73, $w_2 = $ 0x1D und $w_3 = $ 0xD6?

b) Bestimmen Sie für die Codeworten $c_4 = $ 0xA24, $c_5 = $ 0x5D6 und $c_6 = $ 0x141, ob zulässig oder korrigierbar und wenn zulässig oder korrigierbar, den Wert?

Bitnummer	12	11	10	9	8	7	6	5	4	3	2	1
Zuordnung	x_7	x_6	x_5	x_4	q_3	x_3	x_2	x_1	q_2	x_0	q_1	q_0
Kontrollbits												
$w_1=$ 0x73												
$w_2=$ 0x1D												
$w_3=$ 0xD6												
$c_4=$ 0xA24												
$c_5=$ 0x5D6												
$c_6=$ 0x141												

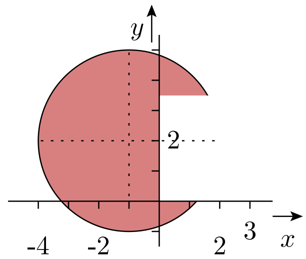
Bitnummer	12	11	10	9	8	7	6	5	4	3	2	1	
Zuordnung	x_7	x_6	x_5	x_4	q_3	x_3	x_2	x_1	q_2	x_0	q_1	q_0	
Kontrollbits													
$w_1=$ 0x73	0	1	1	1	1	0	0	1	1	1	1	0	$c_1=$ 0x79E
$w_2=$ 0x1D	0	0	0	1	1	1	1	0	0	1	1	1	$c_2=$ 0x1E7
$w_3=$ 0xD6	1	1	0	1	1	0	1	1	1	0	0	1	$c_3=$ 0xDB9
$c_4=$ 0xA24	1	0	1	0	0	0	1	0	0	1	0	0	$dq_4 = 3$
$c_5=$ 0x5D6	0	1	0	1	1	1	0	1	0	1	1	0	$dq_5 = 9$
$c_6=$ 0x141	0	0	0	1	0	1	0	0	0	0	0	1	$dq_6 = 15$

3 Formatüberwachung

3.3 Invarianten, WB

Aufgabe 5.6: Kontrollausdruck

Die Wertpaare (x, y) sollen Punkte der im nachfolgenden Bild eingezeichneten Kreisfläche mit dem Mittelpunkt $(-1, 2)$ und dem Radius 3 mit dem ausgeschnittenen rechteckigen Bereich sein.



Entwickeln Sie einen Kontrollausdruck für die Wertebereichskontrolle, der genau dann wahr ist, wenn ein Punkt (x, y) im zulässigen Bereich liegt.

$$((x < 0) \vee (y < 0) \vee (y > 3{,}5)) \wedge \left((x+1)^2 + (y-2)^2 < 3^2\right)$$

3.4 Syntax

Aufgabe 5.7: Kontrollautomat

Ein (vereinfachter) Rechnerbefehlssatz besteht aus vier verschiedenen Befehlstypen

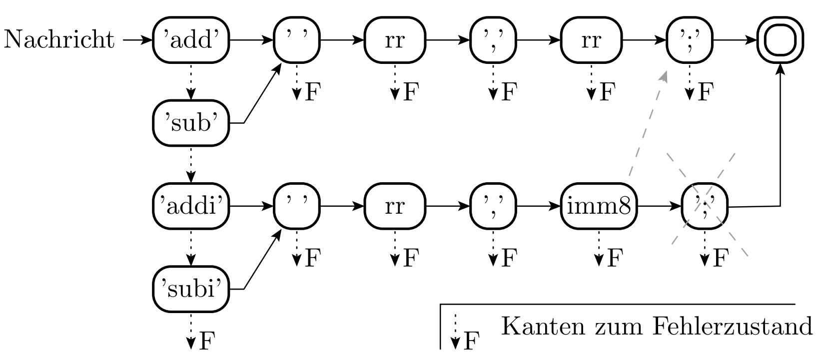
```
add␣rr,rr;
addi␣rr,imm8;
sub␣rr,rr;
subi␣rr,imm8;
```

␣ – Leerzeichen; »rr« Bezeichner eines der 32 Register ("r0", "r1", ... "r31"); »imm8« für die Wert einer 8-Bit Hexzahl ("0x00", "0x01", ..., "0xFF"; "0x" gefolgt von zwei Hex.-Ziffern mit den Zifferenwerten '0' bis 'F').

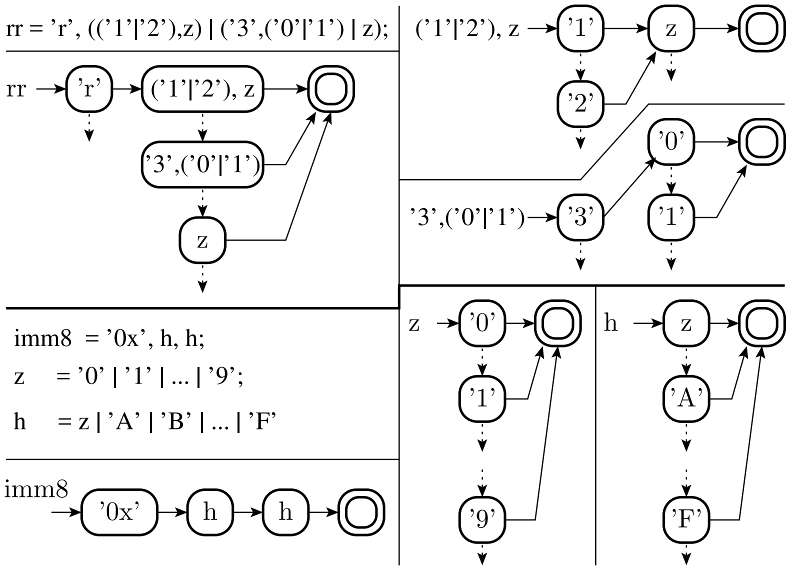
a) Beschreiben Sie das Befehlsformat in der EBNF mit den Ersetzungsregeln für Sequenz, Option, Wiederholung etc.

b) Entwerfen Sie einen deterministischen Kontrollautomaten auf Syntaxfehler als Graph für einen Moore-Automaten.

```
Befehl = ('add' | 'sub', '␣', rr, ',', rr,   ';') |
         ('addi'| 'subi','␣', rr, ',', imm8, ';');
rr     = 'r', (('1'|'2'),z) | ('3',('0'|'1') | z);
imm8   = '0x', h, h; z      = '0' | '1' | ... | '9';
h      = z | 'A' | 'B' | ... | 'F'
```



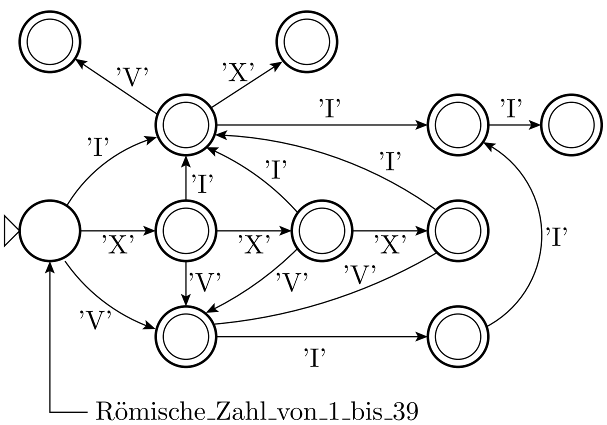
Testautomaten für den Test der Sprachbestandteile:



Aufgabe 5.8: Syntaxtest für römische Zahlen

Entwerfen Sie einen Mealy-Kontrollautomaten[1] für einen Syntaxtest für römische Zahlen mit einem Wert von 1 bis 39.

Wert		Wert		Wert		Wert	
1	I	11	XI	21	XXI	31	XXXI
2	II	12	XII	22	XXII	32	XXXII
3	III	13	XIII	23	XXIII	33	XXXIII
4	IV	14	XIV	24	XXIV	34	XXXIV
5	V	15	XV	25	XXV	35	XXXV
6	VI	16	XVI	26	XXVI	36	XXXVI
7	VII	17	XVII	27	XXVII	37	XXXVII
8	VIII	18	XVIII	28	XXVIII	38	XXXVIII
9	IX	19	XIX	29	XXIX	39	XXXIX
10	X	20	XX	30	XXX		



I	XI	XXI	XXXI
II	XII	XXII	XXXII
III	XIII	XXII	XXXII
IV	XIV	XXIV	XXXIV
V	XV	XXV	XXXV
VI	XVI	XXVI	XXXVI
VII	XVII	XXVII	XXXVII
VIII	XVIII	XXVIII	XXXVIII
IX	XIX	XXIX	XXXIX
X	XX	XX	

(potentieller) Endzustand

Bei allen Eingaben, für die keine Kante gezeichnet ist, Übergang in den Fehlerzustand.

[1] Ein Mealy-Automat, der die Zeichen an den Kanten abräumt.

4 Überwachung auf Richtigkeit

Aufgabe 5.9: Kontrollausdruck

Scheiben Sie einen Testrahmen, den das nachfolgende fehlerhafte C-Programm für die Wurzelberechnung

```
uint8_t wurzel(uint16_t x){
 uint8_t w=0;
 uint16_t sum=0;
 while (sum<x){sum += (w<<1)+1;
 w++;}
 return w;
}
```

mit 1000 zufälligen Werten testet. Ergebniskontrolle mit der inversen Funktion und Fenstervergleich

$$y^2 \leq x < (y+1)^2$$

Protokollierung aller x und y , die die Ergebniskontrolle nicht bestehen. Nutzen Sie dafür »rand()« aus »std_lib.h«.

Zur Kontrolle

```
#include <std_lib.h>
#include <time.h>
#include <stdio.h>
int main(){
 uint16_t x, y, xmin, xmax;
 srand(time(NULL));   // Init. Pseudozufallsg.*
 for (idx=0; idx<1000; idx++){
  x = rand() & 0xFF;  // Begrenzung auf 8 Bit
  y = wurzel(x);      // Testobjekt
  xmin = y*y;         // inversen Fkt.
  xmax = (y+1)*(y+1); // zu Kontrolle
  if ((x<xmin)||(x>xmax)){
   printf("x=%d, y=%d, y^2=%d, (y+1)^2=%d\n",
      x, y, xmin, xmax);
  }
 }
}
```

*time(NULL) liefert Sekunden seit dem 01.01.1970.

Aufgabe 5.10: Vergleichsfenster

Zwei zu vergleichende voneinander unabhängige normalverteilte Zufallsgrößen X_1 und X_2 haben denselben Erwartungswert und die Standardabweichungen $\mathrm{sd}[X_1] = 3$ und $\mathrm{sd}[X_2] = 4$. Wie groß ist für eine Kontrolle

```
if (abs(X1-X2)>eps) {<Fehlerbehandlung>};
```

der Radius ε des Vergleichsfenster mindestens zu wählen, damit die Wahrscheinlichkeit für Vergleichs-Phantom-FF $p_{\mathrm{Phan}} \leq 0{,}1\%$ ist?

$\mathbb{E}[X_1 - X_2] =$ $\varepsilon =$

$\mathrm{sd}[X_1 - X_2] =$

z	...,0	...,1	...,2	...,3	...,4	...,5	...,6	...,7	...,8	...,9
0,...	0,5000	0,5398	0,5793	0,6179	0,6554	0,6915	0,7257	0,7580	0,7881	0,8159
1,...	0,8413	0,8643	0,8849	0,9032	0,9192	0,9332	0,9452	0,9554	0,9641	0,9713
2,...	0,9772	0,9821	0,9861	0,9893	0,9918	0,9938	0,9953	0,9965	0,9974	0,9981
3,...	0,9987	0,9990	0,9993	0,9995	0,9997	0,9998	0,9998	0,9999	0,9999	1,0000

Differenz der Erwartungswerte:

$$\mathbb{E}[X_1 - X_2] = 0$$

Die Varianz der Differenzen ist die Summe der Varianzen:

$$\mathrm{sd}[X_1 - X_2] = \sqrt{\mathrm{Var}[X_1] + \mathrm{Var}[X_2]} = \sqrt{3^2 + 4^2} = 5$$

Standadisierter Normalverteilungswert für beiderseitig $\alpha_1 = \alpha_2 = 0{,}05\%$ ist etwa 3,3.

z	...,0	...,1	...,2	...,3	...,4	...,5	...,6	...,7	...,8	...,9
0,...	0,5000	0,5398	0,5793	0,6179	0,6554	0,6915	0,7257	0,7580	0,7881	0,8159
1,...	0,8413	0,8643	0,8849	0,9032	0,9192	0,9332	0,9452	0,9554	0,9641	0,9713
2,...	0,9772	0,9821	0,9861	0,9893	0,9918	0,9938	0,9953	0,9965	0,9974	0,9981
3,...	0,9987	0,9990	0,9993	0,9995	0,9997	0,9998	0,9998	0,9999	0,9999	1,0000

Mindestintervallradius für das Vergleichsfenster:

$$\varepsilon \approx 3{,}3 \cdot 5 = 16{,}5$$

Aufgabe 5.11: Diversitätsabschätzung

Bei einer Kontrolle durch Verdopplung und Vergleich wurden von $\#FF = 300$ Fehlfunktionen $\#k_{\mathrm{ist}} = 5$ nicht erkannt.

1. Auf welchen Bereich der zu erwartenden Anzahl der nicht erkannten Fehlfunktionen lässt das Experiment schließen? Zulässige Irrtumswahrscheinlichkeiten, dass im Experiment ein Werte oberhalb oder unterhalb des Bereichs hätte auftreten können, $\alpha_1 = \alpha_2 = 10\%$.
2. Auf welchen Bereich der Diversität lässt das Experiment schließen?

Hinweise:

1. Zählwert X ist poisson-verteilt.
2. Schätzwert der zu erwartenden Diversität nach TV-F1, Abschn. 3.2 Überwachungsverfahren:

$$\hat{Div} = \frac{\#DFF}{\#FF} = 1 - \frac{\#k_{\mathrm{ist}}}{\#FF}$$

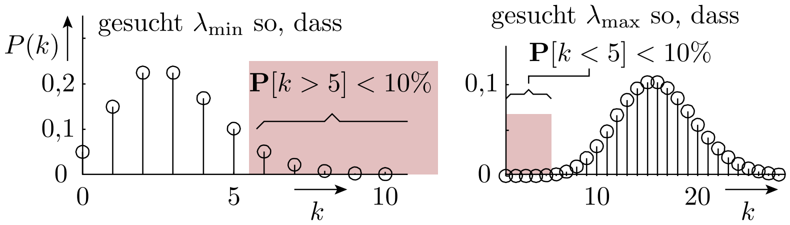
Zur Kontrolle

Von $\#FF = 300$ Fehlfunktionen wurden $x_{\mathrm{ist}} = \#FF_{\mathrm{M}} = 5$ nicht erkannt. Zulässige Irrtumswahrscheinlichkeiten: $\alpha_1 = \alpha_2 = 10\%$.

---

1. Unter- und Obergrenze des zu erwartenden Zählwerts:

   [3,15, 7,99]



$\alpha_1 = \alpha_2$	$k_{\mathrm{ist}} = 4$	$k_{\mathrm{ist}} = 5$	$k_{\mathrm{ist}} = 6$
2%	[1,53, 9,08]	[2,09, 10,6]	[2,68, 12,0]
10%	[2,43, 6,68]	[3,15, 7,99]	[3,89, 9,28]
20%	[3,09, 5,51]	[3,90, 6,73]	[4,73, 7,91]

2. Unter- und Obergrenze der zu erwartenden Diversität:

$$\mathbb{E}\left[Div\right]_{\min} = 1 - \frac{\lambda_{\max}}{\#FF} = 1 - \frac{7{,}99}{300} = 97{,}3\%$$
$$\mathbb{E}\left[Div\right]_{\max} = 1 - \frac{\lambda_{\min}}{\#FF} = 1 - \frac{3{,}15}{300} = 99{,}0\%$$

5 Fehlertoleranz

5.1 Fehlerbehandlung

Aufgabe 5.12: Beispiele für die Fehlerbehandlung

Nennen Sie Beispiele (Ihnen bekannte Programme und Geräte) die folgende Techniken nutzen:

1. Zeitüberwachung mit Service-Abbruch bei Zeitüberschreitung.
2. Wiederholungsanforderung nach fehlerhaftem Datenempfang.
3. Systemen, bei denen sich Fehlverhalten durch andere Eingabereihenfolgen, Nutzung andere Eingabemenüs etc. umgehen lassen.
4. Systeme, die vor dem Ausschalten automatisch ihre Bearbeitungszustand sichern.
5. Systeme, die nach einer Fehlfunktion vom letzten gesicherten Zustand starten.
6. Versenden von Fehlerinformationen an die Firma, die das System entwickelt hat.

Zur Kontrolle

1. Zeitüberwachung mit Abbruch bei Zeitüberschreitung: Lesezugriffe auf Laufwerke. Lesezugriffe auf Daten im Internet. ...
2. Wiederholung nach fehlerhaftem Datenempfang: Standardreaktion auf Prüfsummenfehler beim Datenempfang, Buskollisionen CAN-Bus, Ethernet, ...
3. Beseitigung des Fehlverhalten durch geänderte Eingabereihenfolge: XFig, Textbearbeitung. Beim Löschen vorwärts Programmabsturz, beim Löschen rückwärts kein Absturz.
4. Automatische Sicherung des Bearbeitungszustands beim Ausschalten: Handys, Tablets, ...
5. Start vom letzten gesicherten Zustand: Typisch für Textverarbeitungssysteme.
6. Versenden von Fehlerberichten: Windows, Linux, ...

Aufgabe 5.13: Fail-Safe/-Fast/-Slow

1. Was besagt das Ruhestromprinzip?
2. Eine Software sei so programmiert, dass mit einem Compieler-Schalter zwischen Fail-Fast und Fail-Slow umgeschalten werden kann. Wann wird es wie übersetzt und warum?

1. Das System wird so aufgebaut, dass bei Ausfall der Kontrollfunktion die Notfallbehandlung eingeleitet wird.
2. Fail-Fast für den Test und Probebetrieb, um möglichst viele Probleme zu erkennen und Fehler zu finden. Fail-Slow für den Einsatz, weil so die Zuverlässigkeit höher ist.

Aufgabe 5.14: Fehlerisolation

1. Welche Konzepte dienen in modernen Betriebssystemen zur Fehlerisolation zwischen nebenläufig auf Rechner abzuarbeitenden Prozessen?
2. Welche Hardware-Funktionen stellen dafür moderne Prozessoren zur Verfügung?

1. Fehlerisolationskonzepte:
   - Virtuelle Adressierung, die jedem Prozess nur Zugriff auf eigene Daten erlaubt.
   - Zugriff auf Betriebssystemdienste (Bereitstellung von physikalischem Speicher, Zugriff auf EA-Geräte, ...) über Systemrufe, ...
2. Hardware-Funktionen für die Fehlerisolation:
   - Adressrecheneinheit, TLB (Übersetzungs-Cache zwischen virtuellen und physikalischen Adressen, Cache-Controller, ...;
   - Systemrufe: Software-Interrupts, privelegierte Befehle z.B. zur Umprogrammierung der TLB- und Cache-Speicher, ...

5.2 Redundanz

Aufgabe 5.15: 3-Versionssystem

Für ein 3-Versionssystem mit den Wahrscheinlichkeiten je SL:

- $p_{\mathrm{FF}} = 10^{-5}$ zufällige Fehlfunktion in einem Teilsystem
- $p_{\mathrm{CC}} = 10^{-1}$ wenn die erste SL eine FF ist, sind die beiden anderen dieselbe FF.

wie groß sind unter der Annahme, dass zwei zufällige Verfälschungen praktisch nie übereinstimmen, die Wahrscheinlichkeiten:

1. p_{CCF} für drei durch gemeinsame Usache gleiche FF,
2. $p_{\mathrm{F}i}$ für i gleichzeitige unabhängige FF,
3. p_{F} für mindestens eine FF
4. p_{FT} bedingte Wahrscheinlichkeit für Tolerierung (genau eine FF, wenn mindestesens eine FF),
5. p_{E}, bedingte Wahrscheinlichkeit für Erkennen ohne Tolerierung (mindestens zwei unabhängige FF, wenn mindestesens eine FF).

Zur Kontrolle

Für ein 3-Versionssystem mit den Wahrscheinlichkeiten je SL:

- $p_{\mathrm{FF}} = 10^{-5}$ zufällige Fehlfunktion in einem Teilsystem
- $p_{\mathrm{CC}} = 10^{-1}$ wenn die erste SL eine FF ist, sind die beiden anderen dieselbe FF.

---

1. identische (Common Cause) FF:

$$p_{\mathrm{CCF}} = p_{\mathrm{FF}} \cdot p_{\mathrm{FA}} = 10^{-5} \cdot 10^{-1} = 10^{-6}$$

2. i unabhängige Fehlerfunktion. Die bedingte Wahrscheinlichkeit für nicht-Common-Cause-FF gehorcht dem Versuchsschema der Binomialverteilung:

$$p_{\mathrm{F}i} = (1 - p_{\mathrm{CCF}}) \cdot \binom{3}{i} \cdot p_{\mathrm{FF}}^{i} \cdot (1 - p_{\mathrm{FF}})^{3-i}$$

i	0	1	2	3
$p_{\mathrm{F}i}$	$1 - 3{,}1 \cdot 10^{-5}$	$3 \cdot 10^{-6}$	$3 \cdot 10^{-10}$	10^{-15}

3. mindestens eine FF:

$$p_{\mathrm{F}} = p_{\mathrm{CCF}} + \sum_{i=1}^{3} p_{\mathrm{F}i} = 4 \cdot 10^{-6}$$

4. bedingte Wahrscheinlichkeit für Tolerierung:

$$p_{\mathrm{FT}} = \frac{p_{\mathrm{F1}}}{p_{\mathrm{F}}} = 0{,}75$$

5. bedingte Wahrscheinlichkeit Erkennen ohne Tolerierung:

$$p_{\mathrm{F2}} = p_{\mathrm{FT}} = \frac{p_{\mathrm{F2}} + p_{\mathrm{F2}}}{p_{\mathrm{F}}} = 3 \cdot 10^{-4}$$

5.4 RAID und Backup

Aufgabe 5.16: Zuverlässigkeiterhöhung durch Redundanz

Gegeben ist ein IT-System aus Rechner, Festplatte, Stromversorgung etc. mit folgenen Teilzuverlässigkeiten:

Teilsystem	Rechner	Festplatte	Stromversorgung	sonstiges
Teilzuverlässigkeit	Z_{R}	Z_{FP}	Z_{SV}	Z_{*}
Wert in SL/FF	1000	500	700	2000

1. Welche Gesamtzuverlässigkeit hat das System?

Teilzuverlässigkeit	Z_{R}	Z_{FP}	Z_{SV}	Z_{*}
Wert in SL/FF	1000	500	700	2000

2. Gesamtzuverlässigkeit, wenn die Festplatte durch ein RAID aus zwei Platten vom bisherigen Typ ersetzt wird, und das RAID nur eine Fehlfunktion weitergibt, wenn beide Platten zeitgleich eine Fehlfunktion haben?

$$Z_{\mathrm{ges}} = \frac{1}{\frac{1}{1000} + \frac{1}{500} + \frac{1}{700} + \frac{1}{2000}} = 203\,\frac{\mathrm{SL}}{\mathrm{FF}}$$

Das RAID versagt, wenn beide Platten (gleichzeitig) versagen:

$$\begin{aligned} \frac{1}{Z_{\mathrm{RAID}}} &= 1 - p_{\mathrm{Z.RAID}} = (1 - p_{\mathrm{Z.FP}})^2 = \frac{1}{Z_{\mathrm{FP}}^2} \\ Z_{\mathrm{RAID}} &= 500^2\,\frac{\mathrm{SL}}{\mathrm{NTFF}} \end{aligned}$$

(NTFF – nicht tolerierte FF). Gesamtzuverlässigkeit:

$$Z_{\mathrm{ges}} = \frac{1}{\frac{1}{1000} + \frac{1}{500^2} + \frac{1}{700} + \frac{1}{2000}} = 341\,\frac{\mathrm{SL}}{\mathrm{FF}}$$

Erhöhung um etwa 40 SL/FF.