Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Eidgenössisches Departement des Innern EDI
Bundesamt für Statistik BFS
Abteilung Gesundheit und Soziales

Silvio Hasler, 23. Juni 2017

Plausibilisierungskonzept der Medizinischen Statistik der Krankenhäuser

Version 6.0.0.2

Dieser Bericht ist ausschliesslich in elektronischer Form in Deutsch und Französisch verfügbar:
http://www.bfs.admin.ch

Autor:

Freudiger EDV-Beratung
Zeughausgasse 16
Postfach
3001 Bern

Inhaltsverzeichnis

A. Kodierung

A0. Diagnosen

A00. Hauptdiagnose

A0000 [E] Hauptdiagnose: Angabe fehlt (Statistikfall A, C)

```
DIA[0].KeyLen == 0 && isIn(MB->V0202,2,_AC) && !(MB->hasMN && strcmp(MN->V2201,'0') == 0)
```

A0010 [E] Unzulässiger Hauptdiagnosecode

```
DIA[0].inIndex && strcmp(DIA[0].Data[ICD_VALIDMD],'2') == 0 && *MB->MD->V4801 == '1'
```

A0011 [E] Sternkode nicht zulässig als Hauptdiagnose

```
DIA[0].inIndex && strcmp(DIA[0].Data[ICD_TYP],'2') == 0
```

A0012 [E] Kodes mit !: immer als Nebendiagnose, ausser die äusseren Ursachen (V01!-Y98!)

```
DIA[0].inIndex && isExklDia(DIA[0].Key)
```

A0013 [E] Hauptdiagnose: Äussere Ursachen (ICD-10 Kap. XX) hier nicht zulässig

```
!DIA[0].isNull && (DIA[0].Key[0] == 'V' || DIA[0].Key[0] == 'W' || DIA[0].Key[0] == 'X' || DIA[0].Key[0] == 'Y')
```

A0020 [E] Ein Fall mit Hauptdiagnose Z758 darf nicht fakturiert werden.

```
DIA[0].inIndex && strcmp(DIA[0].Key,'Z758') == 0 && MB->hasMD && strcmp(MB->MD->V4801,'1') == 0
```

A0021 [E] Ein Fall mit Hauptdiagnose Z758 muss Eintrittsart 4 Interner Übertritt haben.

```
DIA[0].inIndex && strcmp(DIA[0].Key,'Z758') == 0 && strcmp(MB->V1203,'4') != 0
```

A01. Zusatz zur Hauptdiagnose

A0110 [E] Kreuzkode nicht zulässig als Hauptdiagnose-Zusatz

```
DIA[1].inIndex && (strcmp(DIA[1].Data[ICD_TYP],'1') == 0 || strcmp(DIA[1].Data[ICD_TYP],'4') == 0)
```

A0120 [E] Zusatz zur Hauptdiagnose: nur für Stern-Kodes und für äussere Ursachen (V01!-Y98!) gültig

```
DIA[1].inIndex && (strcmp(DIA[1].Data[ICD_TYP],'1') == 0 || strcmp(DIA[1].Data[ICD_TYP],'4') == 0)
```

A0130 [W] Hauptdiagnosecodes T90 - T98 benötigen keine Zusatzdiagnose V - Y mehr.

```
DIA[0].inIndex && strncmp(DIA[0].Key,'T9',2) == 0 && DIA[1].inIndex && (strncmp(DIA[1].Key,'V',1) == 0 || strncmp(DIA[1].Key,'W',1)
== 0 || strncmp(DIA[1].Key,'X',1) == 0 || strncmp(DIA[1].Key,'Y',1) == 0)
```

A02. Diagnosen gültigkeit

A0200 [E] Ungültiger Diagnosekode (ICD-10)

!DIA[i].isNull && !DIA[i].inIndex

A0201 [E] ICD-Kode zum Austrittszeitpunkt nicht mehr gültig

DIA[i].inIndex && strlen(DIA[i].Data[ICD_VALID_TO]) > 0 && MB->dateV1501.hasYear && MB->dateV1501.Year > atol(DIA[i].Data[ICD_VALID_TO])

A0202 [E] ICD-Kode zum Austrittszeitpunkt noch nicht gültig

DIA[i].inIndex && strlen(DIA[i].Data[ICD_VALID_FROM]) > 0 && MB->dateV1501.hasYear && MB->dateV1501.Year < atol(DIA[i].Data[ICD_VALID_FROM])

A0210 [E] Nicht terminaler ICD 10-Kode

DIA[i].inIndex && StatYear > 2009 && StatYear >= atol(DIA[i].Data[ICD_VALID_FROM]) && StatYear <= atol(DIA[i].Data[ICD_VALI && (StatYear < atoi(DIA[i].Data[ICD_TERM_FROM]) || StatYear > atoi(DIA[i].Data[ICD_TERM_TO]))

A03. Kodierregeln

A0310 [W] Diagnose / Austrittsart: Letale Hauptdiagnose, als Austrittsart nur Tod möglich

!DIA[i].isNull && strcmp(MB->V1502,'5') != 0 && (strncmp(DIA[i].Key,'R95',3) == 0 || strncmp(DIA[i].Key,'R96',3) == 0 || strncmp(DIA[i].Key,'R98',3) == 0 || strncmp(DIA[i].Key,'R99',3) == 0 || strncmp(DIA[i].Key,'P95',3) == 0 || strncmp(DIA[i].Key,'O95',3) == 0 || strncmp(DIA[i].Key,'O96',3) == 0 || strncmp(DIA[i].Key,'O97',3) == 0 || strncmp(DIA[i].Key,'I461',4) == 0 || strncmp(DIA[i].Key,'P964',4) == 0 || strncmp(DIA[i].Key,'S18',3) == 0)

A0320 [E] Lebendgeborene müssen mit dem Diagnosekode Z38 kodiert werden

strcmp(MB->V0202,'A') == 0 && MB->hasMN && strcmp(MB->V1203,'3') == 0 && strcmp(MB->MN->V2201,'1') == 0 && !hasDiaLike('Z38')

A0321 [E] Vitalstatus (Totgeburt) / Diagnosen: Widersprüchliche Dokumentation eines Lebendgeborenen mit einem Diagnosecode Z38.-

DIA[i].inIndex && strncmp(DIA[i].Key,'Z38',3) == 0 && MB->hasMN && strcmp(MB->MN->V2201,'0') == 0

A0330 [E] Die Kodes R75, Z21, B23.0 und die Gruppe B20-B24 schliessen sich gegenseitig aus und sind während desselben stationären Aufenthaltes nicht zusammen aufzuführen

DIA[i].inIndex && hasB2024 && (strncmp(DIA[i].Key,'R75',3) == 0 || strncmp(DIA[i].Key,'Z21',3) == 0 || strncmp(DIA[i].Key,'B230',4) == 0)

A0341 [E] Der Z37-Code muss im Datensatz der Mutter, nicht in demjenigen des Kindes kodiert werden

strlen(MB->V1103) > 0 && DIA[i].inIndex && strncmp(DIA[i].Key,'Z37',3) == 0 && atoi(MB->V1103) == 0

A0342 [E] Der Z38-Code muss im Datensatz des Kindes, nicht in demjenigen der Mutter kodiert werden

strlen(MB->V1103) > 0 && DIA[i].inIndex && strncmp(DIA[i].Key,'Z38',3) == 0 && atoi(MB->V1103)>0

A0343 [W] Wird einer der folgenden Kodes als Haupt- oder Nebendiagnose angegeben, muss der Kode O09.-! als Nebendiagnose angegeben werden: O03-O07; O10-O16; O20-O29; O30-O48; O60-O75; O80-O82; Z34-Z37

DIA[i].inIndex && isSSDia(DIA[i].Key) && !hasO09

A0344 [E] Wenn der Code O80 - O82 steht, muss Z37 kodiert werden und gehört in die Nebendiagnose

DIA[i].inIndex && (strncmp(DIA[i].Key,'O80',3) == 0 || strncmp(DIA[i].Key,'O81',3) == 0 || strncmp(DIA[i].Key,'O82',3) == 0) && !hasZ37

A0345 [E] Die Codes O80, O81, O82 sind Hauptdiagnose-Codes.

i > 0 && DIA[i].inIndex && (strncmp(DIA[i].Key,'O80',3) == 0 || strncmp(DIA[i].Key,'O81',3) == 0 || strncmp(DIA[i].Key,'O82',3) == 0)

A0350 [W] Die Diagnosen Z51.0, Z51.1, Z51.82 werden nicht kodiert.

DIA[i].inIndex && (strcmp(DIA[i].Key,'Z510') == 0 || strcmp(DIA[i].Key,'Z511') == 0 || strcmp(DIA[i].Key,'Z5182') == 0)

A0361 [X] Bei einem Einling muss einer der Kodes Z38.0/1/2 kodiert werden

*V2201 == '1' && *V2202 == '1' && !(MB->MD->hasDiaLike('Z380') || MB->MD->hasDiaLike('Z381') || MB->MD->hasDiaLike('Z382'))

A0362 [X] Bei einem Zwilling muss einer der Kodes Z38.3/4/5 kodiert werden

*V2201 == '1' && *V2202 == '2' && !(MB->MD->hasDiaLike('Z383') || MB->MD->hasDiaLike('Z384') || MB->MD->hasDiaLike('Z385'))

A0363 [X] Bei einem Mehrling muss einer der Kodes Z38.6/7/8 kodiert werden

*V2201 == '1' && atoi(V2202) > 2 && !(MB->MD->hasDiaLike('Z386') || MB->MD->hasDiaLike('Z387') || MB->MD->hasDiaLike('Z388'))

A0364 [X] Wenn ein Einling mit Z38.0/1/2 kodiert wird, muss er auch im Neugeborenendatensatz deklariert sein

(hasDiaLike('Z380') || hasDiaLike('Z381') || hasDiaLike('Z382')) && (!MB->hasMN || *(MN->V2201) != '1' || *(MN->V2202) != '1')

A0365 [X] Wenn ein Zwilling mit Z38.3/4/5 kodiert wird, muss er auch im Neugeborenendatensatz deklariert sein

(hasDiaLike('Z383') || hasDiaLike('Z384') || hasDiaLike('Z385')) && (!MB->hasMN || *(MN->V2201) != '1' || *(MN->V2202) != '2')

A0366 [X] Wenn ein Mehrling mit Z38.6/7/8 kodiert wird, muss er auch im Neugeborenendatensatz deklariert sein

(hasDiaLike('Z386') || hasDiaLike('Z387') || hasDiaLike('Z388')) && (!MB->hasMN || *(MN->V2201) != '1' || atoi(MN->V2202) < 3)

A04. Diagnosen Plausibilität

A0400 [W] Diagnosekode nicht vereinbar mit Geschlecht

DIA[i].inIndex && strlen(DIA[i].Data[ICD_SEX]) > 0 && strcmp(DIA[i].Data[ICD_SEX],MB->V1101) != 0

A0410 [W] ICD-Kode unvereinbar mit Alter für a) Säugling (hauptsächlich Neugeborene, jedoch auch Patienten zwischen 1 und 364/5 Tagen)

DIA[i].inIndex && strcmp(DIA[i].Data[ICD_AGEGRP],'a') == 0 && MB->AGED > 365

A0411 [W] ICD-Kode unvereinbar mit Alter für b) Kleinkind (Alter zwischen 1 Tag und 2 Jahren)

```
DIA[i].inIndex && strcmp(DIA[i].Data[ICD_AGEGRP],'b') == 0 && (MB->AGEJ >= 2 || MB->AGED == 0)
```

A0412 [W] ICD-Kode unvereinbar mit Alter für c) Kind (Alter zwischen 1 Tag und 10 Jahren)

```
DIA[i].inIndex && strcmp(DIA[i].Data[ICD_AGEGRP],'c') == 0 && (MB->AGEJ >= 10 || MB->AGED == 0)
```

A0413 [W] ICD-Kode unvereinbar mit Alter für d) Kinder und Jugendliche (Alter zwischen 1 Tag und 19 Jahren)

```
DIA[i].inIndex && strcmp(DIA[i].Data[ICD_AGEGRP],'d') == 0 && (MB->AGEJ >= 19 || MB->AGED == 0)
```

A0414 [W] ICD-Kode unvereinbar mit Alter für e) Post-neonatal (älter als 27 Tage)

```
DIA[i].inIndex && strcmp(DIA[i].Data[ICD_AGEGRP],'e') == 0 && MB->AGED < 28
```

A0415 [W] ICD-Kode unvereinbar mit Alter für f) Älter als Säugling (Alter ab 1 Jahr)

```
DIA[i].inIndex && strcmp(DIA[i].Data[ICD_AGEGRP],'f') == 0 && MB->AGEJ < 1
```

A0416 [W] ICD-Kode unvereinbar mit Alter für g) Pubertät (Alter zwischen 8 und 19 Jahren)

```
DIA[i].inIndex && strcmp(DIA[i].Data[ICD_AGEGRP],'g') == 0 && (MB->AGEJ < 8 || MB->AGEJ > 19)
```

A0417 [W] ICD-Kode unvereinbar mit Alter für h) Gebärfähiges Alter (Patientinnen zwischen 12 und 55 Jahren)

```
DIA[i].inIndex && strcmp(DIA[i].Data[ICD_AGEGRP],'h') == 0 && (MB->AGEJ < 12 || MB->AGEJ > 55)
```

A0418 [W] ICD-Kode unvereinbar mit Alter für j) Erwachsene (ab 15 Jahren)

```
DIA[i].inIndex && strcmp(DIA[i].Data[ICD_AGEGRP],'j') == 0 && MB->AGEJ < 15
```

A0419 [W] ICD-Kode unvereinbar mit Alter für k) Reife Erwachsene (ab 30 Jahren)

```
DIA[i].inIndex && strcmp(DIA[i].Data[ICD_AGEGRP],'k') == 0 && MB->AGEJ < 30
```

A05. Diagnose-Attribute

A0510 [W] Dieser ICD-Kode benötigt eine Seitigkeit

```
i != 1 && DIA[i].inIndex && strcmp(DIA[i].Data[ICD_LATERAL],'1') == 0
```

A09. Formell

A0910 [H] Mögliche kongenitiale Missbildung.

```
strlen(DIA[i].Key) > 0 && strncmp(DIA[i].Key,'Q',1) == 0 && strcmp(MB->V0301,'1') == 0
```

A0920 [H] Meldepflichtige Diagnose

```
DIA[i].inIndex && (hasKeyLikeLen(DIA[i].Key,4,'|A392|A393|A394|A395|A398|A399|A403|A413|A481|A482|A491|A492|A810|A841|B160|B
|| hasKeyLikeLen(DIA[i].Key,3,'|J09|J13|J14|'))
```

A0990 [X] Die Diagnose ist weiter oben schon vorhanden (Duplikat)

strlen(DIA[i].Key) > 0 && strcmp(DIA[j].Key, DIA[i].Key) == 0

A1. Behandlungen

A10. Hauptbehandlung

A1000 [W] Hauptbehandlung: Angabe fehlt

BEH[0].isNull && isIn(MB->V0202,2,_AC) && !isIn(MB->V1401,7,_7_HKSTH) && strncmp(DIA[0].Key,'Z38',3) != 0 && strncmp(DIA[0].Key,'Z758',4) != 0

A1001 [H] Hauptbehandlung: Angabe fehlt

BEH[0].isNull && isIn(MB->V0202,2,_AC) && isIn(MB->V1401,7,_7_HKSTH) && strcmp(MB->V1401,'M200') != 0 && strncmp(DIA[0].Key,'Z38',3) != 0 && strncmp(DIA[0].Key,'Z758',4) != 0

A1005 [E] Zusatzkodes sind keine Hauptbehandlungen

*BEH[0].Data[CHOP_VALIDHB] == '2'

A12. Gültigkeit

A1200 [E] Ungültiger Behandlungskode (CHOP)

!BEH[i].isNull && !BEH[i].inIndex

A1201 [E] CHOP-Kode zum Austrittszeitpunkt nicht mehr gültig

BEH[i].inIndex && strlen(BEH[i].Data[CHOP_VALID_TO]) > 0 && MB->dateV1501.hasYear && MB->dateV1501.Year > atol(BEH[i].Data[CHOP_VALID_TO])

A1202 [E] CHOP-Kode zum Austrittszeitpunkt noch nicht gültig

BEH[i].inIndex && strlen(BEH[i].Data[CHOP_VALID_FROM]) > 0 && MB->dateV1501.hasYear && MB->dateV1501.Year < atol(BEH[i].Data[CHOP_VALID_FROM])

A1210 [E] Nicht terminaler CHOP-Kode

BEH[i].inIndex && StatYear > 2009 && StatYear >= atol(BEH[i].Data[CHOP_VALID_FROM]) && StatYear <= atol(BEH[i].Data[CHOP_V && (StatYear < atoi(BEH[i].Data[CHOP_TERM_FROM]) || StatYear > atoi(BEH[i].Data[CHOP_TERM_TO]))

A1222 [E] Unspezifischer 2-stelliger Behandlungskode (CHOP)

!BEH[i].inIndex && strcmp(BEH[i].Data[CHOP_TYP],'2') == 0

A1223 [E] Unspezifischer 3-stelliger Behandlungskode (CHOP)

BEH[i].inIndex && strcmp(BEH[i].Data[CHOP_TYP],'3') == 0

A13. Kodierregeln

A1310 [W] Behandlungskode nur bei Todesfall möglich

strcmp(MB->V1502,'5') != 0 && strncmp(BEH[i].Key,'8981',4) == 0

A14. Plausibilität

A1400 [W] Behandlungskode nicht vereinbar mit Geschlecht

BEH[i].inIndex && strlen(BEH[i].Data[CHOP_SEX]) > 0 && strcmp(BEH[i].Data[CHOP_SEX],MB->V1101) != 0 && !hasF64

A1401 [E] Behandlungscode aus Kapitel 13 unvereinbar mit Geschlecht (Mann)

strcmp(MB->V0202,'A') == 0 && BEH[i].inIndex && strcmp(MB->V1101,'1') == 0 && strncmp(BEH[i].Key,'72',2) >= 0 && strncmp(BEH[i].Key,'75',2) <= 0

A1410 [W] Kode nicht vereinbar mit Alter (nur für Peri-/Neonatalperiode verwendbar)

DIA[i].inIndex && DIA[i].Key[0] == 'P' && MB->dateV1201.DateValue - MB->dateV1102.DateValue > 28 && (DIA[i].Key[1] == '1' || isSubKeyIn(DIA[i].Key,1,2,16,_16_DIA_PERINAT_LEN2) || isSubKeyIn(DIA[i].Key,1,3,10,_10_DIA_PERINAT_LEN3))

A1412 [W] Behandlungscode aus Kapitel 13 unvereinbar mit Alter (< 12 Jahre)

strcmp(MB->V0202,'A') == 0 && BEH[i].inIndex && atoi(MB->V1103) < 12 && strncmp(BEH[i].Key,'72',2) >= 0 && strncmp(BEH[i].Key,'75',2) <= 0

A1417 [W] CHOP-Kode unvereinbar mit Alter für i) Gebärfähiges Alter (Patientinnen zwischen 12 und 55 Jahren)

BEH[i].inIndex && strcmp(BEH[i].Data[CHOP_AGEGRP],'i') == 0 && (MB->AGEJ < 12 || MB->AGEJ > 55)

A18. Behandlungsdatum

A1800 [E] Behandlungsdatum: Angabe fehlt

!BEH[i].isNull && BEH[i].dateDatum.isNull

A1810 [E] Behandlungsdatum: Ungültige Datumsangabe

!BEH[i].dateDatum.isNull && !BEH[i].dateDatum.hasDate

A1811 [E] Behandlungsdatum: Ungültige Stundenangabe

BEH[i].dateDatum.hasDate && !BEH[i].dateDatum.hasHour && (BEH[i].dateDatum.Length > 8 || i == 0)

A1815 [E] Nebenbehandlungen werden ohne Stunde übermittelt

i > 0 && BEH[i].dateDatum.hasDate && strlen(BEH[i].dateDatum.Value) > 8

A1821 [E] MD: Eintritt darf nicht später als Behandlung sein

strcmp(MB->V0202,'A') == 0 && (MB->dateV1201.hasDate && BEH[i].dateDatum.hasDate && MB->dateV1201.DateValue > BEH[i].dateDatum.DateValue) || (MB->dateV1201.hasHour && BEH[i].dateDatum.hasHour && MB->dateV1201.DateValue == BEH[i].dateDatum.DateValue && MB->dateV1201.Hour > BEH[i].dateDatum.Hour)

A1822 [E] MD: Behandlung darf nicht später als Austritt sein

(MB->dateV1501.hasDate && BEH[i].dateDatum.hasDate && MB->dateV1501.DateValue < BEH[i].dateDatum.DateValue) || (MB->dateV1501.hasHour && BEH[i].dateDatum.hasHour && MB->dateV1501.DateValue == BEH[i].dateDatum.DateValue && MB->dateV1501.Hour < BEH[i].dateDatum.Hour)

A1825 [E] Geburtsdatum darf nicht später als Beginn der Hauptbehandlung sein

dateV1102.hasYear && dateV1702.hasDate && dateV1102.DateValue > dateV1702.DateValue

A19. Formell

A1910 [X] Die Lokalisation der Stents muss vor dem Kode 00.4A.xx angegeben werden

BEH[i].inIndex && strncmp(BEH[i].Key,'004A',4) == 0 && (i == 0 || !BEH[i-1].inIndex || BEH[i-1].KeyLen != 6 || strstr('|397211|397221|397928|397921|397922|397924|397925|397926|397927|397929|399931|399932|',BEH[i-1].Key) == NULL)

A1911 [X] Die Lokalisation der Stents muss am gleichen Tag wie der Kode 00.4A.xx gemacht werden

BEH[i].inIndex && strncmp(BEH[i].Key,'004A',4) == 0 && i > 0 && BEH[i-1].inIndex && BEH[i-1].KeyLen == 6 && strstr('|397211|397221|397928|397921|397922|397924|397925|397926|397927|397929|399931|399932|',BEH[i-1].Key) != NULL && BEH[i].dateDatum.DateValue != BEH[i-1].dateDatum.DateValue

A1920 [X] Unpräziser Kode - darf nicht verwendet werden

strcmp(BEH[i].Key, '399530') == 0 || strcmp(BEH[i].Key, '399540') == 0 || strcmp(BEH[i].Key, '399570') == 0 || strcmp(BEH[i].Key, '399580') == 0 || strcmp(BEH[i].Key, '3995A0') == 0 || strcmp(BEH[i].Key, '3995C0') == 0 || strcmp(BEH[i].Key, '3995D0') == 0 || strcmp(BEH[i].Key, '3995E0') == 0

A1990 [X] Die Behandlung ist weiter oben schon vorhanden (Duplikat)

strlen(BEH[i].Key) > 0 && strcmp(BEH[j].Key, BEH[i].Key) == 0

B. Administrative Daten

B0. Patientendaten

B0100 [E] Geschlecht: Angabe fehlt

strlen(V1101) == 0

B0101 [E] Geschlecht: Ungültige Angabe

strlen(V1101) > 0 && !isIn(V1101,2,_12)

B0110 [E] Geburtsdatum: Angabe fehlt

dateV1102.isNull

B0111 [E] Geburtsdatum: Ungültige Angabe

!dateV1102.isNull && !(dateV1102.isYYYYMMDD || dateV1102.isYYYY0000)

B0115 [E] Geburtsdatum darf nicht später als Eintrittsdatum sein

dateV1102.hasYear && dateV1201.hasDate && dateV1102.DateValue > dateV1201.DateValue

B0120 [E] Alter bei Eintritt: Angabe fehlt

strlen(V1103) == 0

B0121 [E] Alter bei Eintritt: Ausserhalb des gültigen Wertebereichs

strlen(V1103) > 0 && (atol(V1103) < 0 || atol(V1103) > 120)

B0122 [E] Geburt (Kind)/Altersangabe: Ein Neugeborenes hat das Alter 0

strcmp(V1203,'3') == 0 && atol(V1103) != 0

B0130 [E] Wohnort: Angabe fehlt

strlen(V1104) == 0

ACTION: V1104="XXXX";

B0131 [E] Wohnort: Ungültige Angabe

!(V1104_regio.inIndex || V1104_plz.inIndex || V1104_wregio.inIndex || V1104_lregio.inIndex || V1104_natio.inIndex)

B0132 [W] Wohnort: Bei Wohnort in der Schweiz Region (oder allenfalls PLZ) angeben

strcmp(V1104,'CHE') == 0

B0140 [E] Nationalität: Angabe fehlt

strlen(V1105) == 0

ACTION: V1105="XXX";

B0141 [E] Nationalität: Ungültige Angabe

!V1105_natio.isNull && !(V1105_natio.inIndex || V1105_wregio.inIndex)

B0200 [E] Geburtsjahr: Liegt vor mehr als 120 Jahren

dateV1102.hasYear && dateV1102.Year < MX->dateV0004.Year - 120

B0201 [E] Geburtsdatum/Eintrittsdatum: Nicht konsistent mit Altersangabe

dateV1201.hasYear && dateV1102.hasYear && strlen(V1103) > 0 && !(dateV1201.Year - dateV1102.Year == atol(V1103) || dateV1201.Year - dateV1102.Year == atol(V1103)+1)

B0202 [E] Geburtsdatum bei Kindern: Bei Kindern bis 2 Jahren muss das vollständige Geburtsdatum angegeben werden

isInt(V1103) && atoi(V1103) < 2 && dateV1102.isYYYY0000

B0203 [E] Geburtsdatum bei Todesfall: Bei Todesfällen muss das vollständige Geburtsdatum angegeben werden

strcmp(V1502,'5') == 0 && dateV1102.isYYYY0000

B0300 [W] Gestationsalter_12M: Angabe fehlt

MB->AGED < 365 && strlen(V4816) == 0 && !MB->hasMN

B0301 [W] Gestationsalter_12M: Unwahrscheinliche Anzahl Wochen

strlen(V4816) == 3 && ((atol(V4816)/10) < 16 || (atol(V4816)/10) > 44)

B0302 [E] Gestationsalter_12M: Ungültige Anzahl Tage

strlen(V4816) == 3 && strchr('0123456',V4816[2]) == NULL

B0305 [W] Gestationsalter_12M: Ungültiges Format

strlen(V4816) > 0 && ((!isHex(V4816) && strlen(V4816) == 16) || (strlen(V4816) != 3 && strlen(V4816) != 16))

B1. Spitalaufenthalt

B10. Eintritt

B1000 [E] Eintrittsdatum: Angabe fehlt

dateV1201.isNull

B1001 [E] Eintrittsdatum: Ungültige Datumsangabe

!dateV1201.isNull && !dateV1201.hasDate

ACTION: V1201=;

B1002 [E] Eintrittsstunde: Ungültige Stundenangabe

dateV1201.Length>8 && dateV1201.hasDate && !dateV1201.hasHour

B1005 [E] Notfalleintritt: Eintrittsdatum muss inklusive Stundenangabe erfolgen

V1203[0] == '1' && dateV1201.hasDate && !dateV1201.hasHour

B1010 [E] Aufenthalt vor Eintritt: Angabe fehlt

strlen(V1202) == 0

ACTION: V1202="9";

B1011 [E] Aufenthalt vor Eintritt: Ungültige Angabe

strlen(V1202) > 0 && !isIn(V1202,9,_9_AUFVOR)

ACTION: V1202="9";

B1020 [E] Eintrittsart: Angabe fehlt

strlen(V1203) == 0

ACTION: V1203="9";

B1021 [E] Eintrittsart: Ungültige Angabe

strlen(V1203) > 0 && !isIn(V1203,7,_7_1234589)

ACTION: V1203="9";

B1022 [E] Eine Mutter darf nie Eintrittsart Geburt haben

strlen(MB->V1103) > 0 && atoi(MB->V1103)>0 && strcmp(MB->V1203,'3') == 0

B1023 [W] Eintritts-, Geburtsdatum, Eintrittsart: Bei Eintritt durch Geburt Eintrittsart überprüfen

strcmp(MB->V1103,'0') == 0 && MB->dateV1102.DateValue == MB->dateV1201.DateValue && strcmp(MB->V1203,'3') != 0 && !hasZ38_147

B1030 [E] Einweisende Instanz: Angabe fehlt

strlen(V1204) == 0

ACTION: V1204="9";

B1031 [E] Einweisende Instanz: Ungültige Angabe

strlen(V1204) > 0 && !isIn(V1204,8,_8_INSTANZ)

ACTION: V1204="9";

B1055 [H] Eintrittsdatum / Beginn Meldeperiode: Liegt mehr als 2 Jahre vor dem Beginn der Meldeperiode

dateV1201.hasDate && MX->dateV0006.hasDate && MX->dateV0006.Year - dateV1201.Year > 2

B1060 [E] MB: Eintritt darf nicht später als Behandlung sein

strcmp(MB->V0202,'A') == 0 && (dateV1201.hasDate && dateV1702.hasDate && dateV1201.DateValue > dateV1702.DateValue) || (dateV1201.hasHour && dateV1702.hasHour && dateV1201.DateValue == dateV1702.DateValue && dateV1201.Hour > dateV1702.Hour)

B1061 [E] Eintrittszeit darf nicht später als Behandlungszeitpunkt sein

dateV1201.hasHour && dateV1702.hasHour && dateV1201.DateValue == dateV1702.DateValue && dateV1201.Hour > dateV1702.Hour

B1065 [E] Eintrittsdatum darf nicht später als Austrittsdatum sein

dateV1201.hasDate && dateV1501.hasDate && dateV1201.DateValue > dateV1501.DateValue

B1066 [E] Eintrittszeit darf nicht später als Austrittszeitpunkt sein

dateV1201.hasHour && dateV1501.hasHour && dateV1201.DateValue == dateV1501.DateValue && dateV1201.Hour > dateV1501.Hour

B11. Aufenthalt

B1110 [E] Behandlungsart: Angabe fehlt

strlen(V1301) == 0

ACTION: V1301="9";

B1111 [E] Behandlungsart: Ungültige Angabe

strlen(V1301) > 0 && !isIn(V1301,4,_4_BEHART)

ACTION: V1301="9";

B1112 [W] Behandlungsart: Unplausible Angabe

strcmp(V1301,'9') == 0

B1113 [E] Behandlungsart: teilstationär ist ab 2009 nicht mehr gültig

strlen(V1301) == 1 && V1301[0] == '2' && dateV1501.Year > 2008 && !(strcmp(V1401,'M500') == 0 && strcmp(V0104,'BE') == 0)

ACTION: doWriteRec=FALSE;

B1115 [H] Aufenthaltsdauer bei ambulanten Fällen: Ungültige Aufenthaltsdauer für ambulante Behandlung

strcmp(V1301,'1') == 0 && dateV1201.hasDate && dateV1501.hasDate && dateV1501.DateValue - dateV1201.DateValue > 1

B1120 [E] Klasse: Angabe fehlt

strlen(V1302) == 0

ACTION: V1302="9";

B1121 [E] Klasse: Ungültige Angabe

strlen(V1302) > 0 && !isIn(V1302,4,_4_KLASSE)

ACTION: V1302="9";

B1122 [H] Klasse: Unplausible Angabe

strcmp(V1301,'3') == 0 && strcmp(V1302,'9') == 0

B1130 [E] Hauptkostenstelle: Angabe fehlt

strlen(V1401) == 0

B1131 [E] Hauptkostenstelle: Ungültige Angabe

strlen(V1401) > 0 && !((StatYear >= 2016 && isIn(V1401,16,_16_KST)) || (StatYear < 2016 && isIn(V1401,14,_14_KST)))

B1135 [W] Hauptkostenstelle / Alter: Obere Altersgrenze überschritten für Hauptkostenstelle Pädiatrie (M400)

strlen(V1103) > 0 && strcmp(V1401,'M400') == 0 && atol(V1103) > 18

B1136 [W] Hauptkostenstelle / Alter: Unterhalb Altersgrenze für Geriatrie (M900)

strlen(V1103) > 0 && strcmp(V1401,'M900') == 0 && atol(V1103) < 30

B1140 [E] Hauptkostenträger: Angabe fehlt

strlen(V1402) == 0

ACTION: V1402="9";

B1141 [E] Hauptkostenträger: Ungültige Angabe

strlen(V1402) > 0 && !isIn(V1402,7,_1234589)

ACTION: V1402="9";

B1150 [W] Administrativer Urlaub, Ferien: Angabe fehlt

strlen(V1304) == 0

B1151 [E] Administrativer Urlaub, Ferien: Ungültige Angabe

strlen(V1304) > 0 && (atol(V1304) < 0 || atol(V1304) > 9999)

ACTION: V1304=;

B1152 [E] Administrativer Urlaub, Ferien: Angabe übersteigt die Dauer des gesamten Aufenthaltes

dateV1501.hasDate && dateV1201.hasDate && atol(V1304) > 0 && (dateV1501.DateValue >dateV1201.DateValue) && atol(V1304) > (dateV1501.DateValue - dateV1201.DateValue + 1) * 24

B12. Austritt

B1200 [E] Austrittsdatum: Angabe fehlt (Statistikfall A)

strlen(V1501) == 0 && V0202[0] == 'A'

B1201 [E] Austrittsdatum: Ungültige Datumsangabe

!dateV1501.isNull && !dateV1501.hasDate

B1202 [E] Austrittsstunde: Ungültige Stundenangabe

dateV1501.Length > 8 && dateV1501.hasDate && !dateV1501.hasHour

B1210 [E] Entscheid für Austritt: Angabe fehlt

strlen(V1502) == 0 && strcmp(V0202,'A') == 0

ACTION: V1502="9";

B1211 [E] Entscheid für Austritt: Ungültige Angabe

strlen(V1502) > 0 && !isIn(V1502,7,_1234589)

ACTION: V1502="9";

B1220 [E] Aufenthalt nach Austritt: Angabe fehlt

strlen(V1503) == 0 && V0202[0] == 'A'

ACTION: V1503="9";

B1221 [E] Aufenthalt nach Austritt: Ungültige Angabe

strlen(V1503) > 0 && !isIn(V1503,10,_0123456789)

ACTION: V1503="9";

B1230 [E] Behandlung nach Austritt: Angabe fehlt

strlen(V1504) == 0 && strcmp(V0202,'A') == 0

ACTION: V1504="9";

B1231 [E] Behandlung nach Austritt: Ungültige Angabe

strlen(V1504) > 0 && !isIn(V1504,8,_01234589)

ACTION: V1504="9";

B1250 [W] Todesfall: Aufenthalt und Behandlung nach müssen beide 0 sein

strcmp(V1502,'5') == 0 && (strcmp(V1503,'0') != 0 || strcmp(V1504,'0') != 0)

B1251 [E] Austrittsdatum bei Todesfall: Angabe muss inklusive Stunde erfolgen

strcmp(V1502,'5') == 0 && !dateV1501.hasHour

B1252 [E] Aufenthalt/Behandlung nach: Angabe nur bei Todesfällen möglich

(strcmp(V1503,'0') == 0 || strcmp(V1504,'0') == 0) && strcmp(V1502,'5') != 0

B1300 [W] Der DRG-Status muss erfasst werden

strcmp(MB->V0202,'A') == 0 && strlen(V4801) == 0

B1301 [E] DRG Status: Ungültige Angabe

strlen(V4801) > 0 && !isIn(V4801,2,_01)

B1310 [W] Wartepatienten werden nicht über DRG abgerechnet

strcmp(DIA[0].Key, 'Z758') == 0 && *(MB->V1203) == '4' && *MB->MD->V4801 == '1'

B2. IPS

B2000 [W] Aufenthalt in Intensivmedizin: Angabe fehlt

strcmp(MB->V0202,'A') == 0 && strlen(V1303) == 0

B2001 [E] Aufenthalt in Intensivmedizin: Ungültige Angabe

strlen(V1303) > 0 && (atol(V1303) < 0 || atol(V1303) > 9999)

ACTION: V1303="9999";

B2010 [W] Im MB-Datensatz sind IPS-Stunden vorhanden. Ein Teil der Felder MD-Intensivmedizin muss ausgefüllt sein

strlen(MB->V1303) > 0 && atol(MB->V1303) != 0 && (strlen(V4401) == 0 || strlen(V4402) == 0) && strcmp(MB->V0202, 'A') == 0

B2011 [E] Der Fall weist IPS-Stunden auf, aber das Feld NEMS ist leer

strlen(MB->V1303) > 0 && atol(MB->V1303) != 0 && (strlen(V4404) == 0 || !(atol(V4404) > 0))

B2110 [E] Die Dauer der künstlichen Beatmung darf nicht kleiner als 0 oder grösser als 99999 Stunden sein

strlen(V4401) > 0 && (atol(V4401) < 0 || atol(V4401) > 99999)

B2120 [H] Die Dauer der künstlichen Beatmung darf nicht grösser als diejenige der IPS-Stunden sein

strcmp(MB->V0202,'A') == 0 && strlen(V4401) > 0 && (atol(V4401) > (atol(MB->V1303)+15)) && atol(MB->V1303) > 0

B2130 [E] Die Dauer der künstlichen Beatmung darf nicht grösser als die Aufenthaltsdauer sein

strcmp(MB->V0202,'A') == 0 && strlen(V4401) > 0 && atol(V4401) > (MB->STAYH+15)

B2210 [E] Der Schweregrad der akuten Erkrankung darf nicht kleiner als 0 oder grösser als 999 Punkte sein

strlen(V4402) > 0 && (atol(V4402) < 0 || atol(V4402) > 999)

B2220 [E] Wenn ein Schweregrad angegeben wurde, so muss auch die Art des Scores eingetragen sein

strlen(V4402) > 0 && strlen(V4403) == 0

B2221 [E] Ungültige Art des Scores

strlen(V4403) > 0 && !isIn(V4403,3,_3_SCOREART)

B2225 [E] Wenn die Art des Scores angegeben wurde, so muss auch ein Schweregrad eingetragen sein

strlen(V4403) > 0 && strlen(V4402) == 0

B2310 [E] Das Total aller NEMS-Schichten darf nicht kleiner als 0 oder grösser als 999999 sein

strlen(V4404) > 0 && (atol(V4404) < 0 || atol(V4404) > 999999)

B2400 [E] Aufenthalt in Intensivmedizin: Angabe übersteigt die Dauer des gesamten Aufenthaltes

dateV1501.hasDate && dateV1201.hasDate && atol(V1303) > 0 && (dateV1501.DateValue >dateV1201.DateValue) && atol(V1303) > (dateV1501.DateValue - dateV1201.DateValue + 1) * 24

B3. Neugeborenendaten

B30. Angaben zur Mutter

B3000 [E] Geburtsdatum der Mutter: Angabe fehlt

strlen(V2301) == 0

B3001 [E] Geburtsdatum der Mutter: Ungültige Datumsangabe (JJJJMMTT)

strlen(V2301) > 6 && dateV2301.hasMonth && (strlen(V2301) != 8 || (dateV2301.Day > 0 && !dateV2301.isYYYYMMDD))

B3002 [E] Geburtsdatum der Mutter: Ungültige Angabe für die Jahrzahl

!dateV2301.isNull && (!dateV2301.hasYear || dateV2301.Year < 1900)

B3003 [E] Geburtsdatum der Mutter: Fehlende/falsche Monatsangabe

!dateV2301.isNull && !dateV2301.hasMonth

B3005 [E] Das Geburtsdatum der Mutter darf nicht dem Geburtsdatum des Kindes entsprechen

strlen(V2301) > 0 && strncmp(V2301,MB->V1102,6) == 0

B3010 [W] Gestationsalter 1: Angabe fehlt

strlen(V2302) == 0

B3011 [E] Gestationsalter 1: Ungültige Anzahl Tage

strlen(V2302) > 0 && (strlen(V2302) != 3 || strchr('0123456',V2302[2]) == NULL)

B3012 [E] Gestationsalter 1: Ungültige Anzahl Wochen

strlen(V2302) > 0 && ((atol(V2302)/10) < 16 || (atol(V2302)/10) > 44)

B3015 [W] Gestationsalter 1: Unwahrscheinliche Anzahl Wochen

strlen(V2302) > 0 && ((atol(V2302)/10) < 16 || (atol(V2302)/10) > 44)

B3020 [W] Gestationsalter 2: Angabe fehlt

strlen(V2303) == 0

B3021 [E] Gestationsalter 2: Ungültige Anzahl Tage

strlen(V2303) > 0 && (strlen(V2303) != 3 || strchr('0123456',V2303[2]) == NULL)

B3022 [E] Gestationsalter 2: Ungültige Anzahl Wochen

strlen(V2303) > 0 && ((atol(V2303) / 10) < 16 || (atol(V2303) / 10) > 45)

B3025 [W] Gestationsalter 2: Unwahrscheinliche Anzahl Wochen

strlen(V2303) > 0 && ((atol(V2303) / 10) < 16 || (atol(V2303) / 10) > 45)

B3026 [W] Gestationsalter 2 ist mehr als 4 Wochen grösser als Gestationsalter 1

strlen(V2302) > 0 && strlen(V2303) > 0 && ((atol(V2303) / 10) - (atol(V2302) / 10) > 4)

B3040 [H] Anzahl vorausg. Schwangerschaften insgesamt: Angabe fehlt

strlen(V2304) == 0

B3041 [E] Anzahl vorausg. Schwangerschaften insgesamt: Ausserhalb des gültigen Wertebereichs

strlen(V2304) > 0 && (atol(V2304) < 0 || atol(V2304) > 40)

B3050 [H] Anzahl vorausg. Lebendgeburten: Angabe fehlt

strlen(V2305) == 0

B3051 [E] Anzahl vorausg. Lebendgeburten: Ausserhalb des gültigen Wertebereichs

strlen(V2305) > 0 && (atol(V2305) < 0 || atol(V2305) > 20)

B3060 [H] Anzahl vorausg. Fehl- oder Totgeburten: Angabe fehlt

strlen(V2306) == 0

B3061 [E] Anzahl vorausg. Fehl- oder Totgeburten: Ausserhalb des gültigen Wertebereichs

strlen(V2306) > 0 && (atol(V2306) < 0 || atol(V2306) > 20)

B3065 [W] Neugeborenen-Zusatzdaten: Inkonsistente Angaben betreffend Schwangerschaften

strlen(V2305) > 0 && strlen(V2306) > 0 && strlen(V2307) > 0 && atol(V2304) > (atol(V2305) + atol(V2306) + atol(V2307))

B3070 [H] Anzahl vorausg. Schwangerschaftsabbrüche: Angabe fehlt

strlen(V2307) == 0

B3071 [E] Anzahl vorausg. Schwangerschaftsabbrüche: Ausserhalb des gültigen Wertebereichs

strlen(V2307) > 0 && (atol(V2307) < 0 || atol(V2307) > 20)

B3080 [E] Verlegung Mutter aus anderem Spital: Angabe fehlt

strlen(V2308) == 0

B3081 [E] Verlegung Mutter aus anderem Spital: Ungültige Angabe

strlen(V2308) > 0 && !isIn(V2308,2,_01)

ACTION: V2308=;

B31. Angaben zum Kind

B3100 [E] Neugeborenen-Zusatzdaten: Fehlender Zusatzdatensatz

strcmp(V0202,'A') == 0 && strcmp(V0301,'1') == 0 && !hasMN

ACTION: V0301="0";

B3101 [E] Ein im Betrieb geborenes Kind muss immer einen MN-Zusatzdatensatz enthalten

*MB->V0202 == 'A' && *MB->V1203 == '3' && *MB->V1103 == '0' && !MB->hasMN

B3102 [H] Neugeborenen-Zusatz / Hauptkostenstelle: Neugeborene sind in der Regel auf der Geburtsabteilung (M300, M400)

strcmp(V2101,'MN') == 0 && !isIn(MB->V1401,2,_2_KST_MN)

B3104 [E] Neugeborenen-Zusatz / Alter bei Eintritt: Die Altersangabe bei Neugeborenen kann nur 0 sein

strcmp(V2101,'MN') == 0 && atol(MB->V1103) != 0

B3105 [E] Vitalstatus / Austrittsart: Angaben stimmen nicht überein

strcmp(V2201,'0') == 0 && strcmp(MB->V1502,'5') != 0

B3110 [H] Interne Geburtsnummer: Angabe fehlt

strlen(V2102) == 0

ACTION: V2102="0";

B3111 [E] Interne Geburtsnummer: Ausserhalb des gültigen Wertebereichs

strlen(V2102) > 0 && atol(V2102) > 9999

ACTION: V2102="0";

B3130 [E] Vitalstatus: Angabe fehlt

strlen(V2201) == 0

B3131 [E] Vitalstatus: Ungültige Angabe

strlen(V2201) > 0 && !isln(V2201,2,_01)

B3140 [E] Mehrling: Angabe fehlt

strlen(V2202) == 0

B3141 [E] Mehrling: Ungültige Angabe

strlen(V2202) > 0 && !isln(V2202,9,_9_MEHRLING)

B3150 [E] Geburtsrang bei Mehrlingsgeburten: Angabe fehlt oder ist nicht korrekt für Mehrlingsschwangerschaft

atol(V2202) > 1 && (strlen(V2203) == 0 || atol(V2203) > atol(V2202))

B3151 [H] Bei einem Einling sollte der Geburtsrang 1 angegeben werden

strcmp(V2202,'1') == 0 && strlen(V2203) > 0 && strcmp(V2203,'1') != 0

ACTION: V2203="1";

B3152 [H] Geburtsrang bei Mehrlingsgeburten: Ungültige Angabe

strlen(V2203) > 0 && !isln(V2203,9,_9_GEBRANG)

ACTION: V2203="1";

B3160 [E] Geburtsgewicht (gr): Angabe fehlt

strlen(V2204) == 0

B3161 [E] Geburtsgewicht (gr): Ausserhalb des gültigen Wertebereichs

strlen(V2204) > 0 && (atol(V2204) < 0 || atol(V2204) > 7000)

B3162 [W] Geburtsgewicht (gr): Unplausible Angabe für ein Lebendgeborenes

strlen(V2204) > 0 && strcmp(V2201,'1') == 0 && atol(V2204) <= 500

B3163 [W] Geburtsgewicht (gr)/Gestationsalter 1: Unplausible Angabe für ein Totgeborenes

strlen(V2204) > 0 && strcmp(V2201,'0') == 0 && (((atol(V2302) <= 217) && (atol(V2204) > 500)) || ((atol(V2302) > 217) && (atol(V2204) <= 500)))

B3164 [W] Geburtsgewicht (gr): Unplausible Angabe

strlen(V2204) > 0 && atol(V2204) <= 7000 && atol(V2204) >= 6000

B3165 [H] Aufnahmegewicht: Angabe fehlt

strlen(V4501) == 0 && strcmp(MB->V1203,'3') == 0

B3166 [E] Bei bis zu 28 Tage alten Säuglingen muss das Aufnahmegewicht erfasst werden.

strcmp(MB->V0202,'A') == 0 && MB->AGED < 28 && strlen(V4501) == 0

B3167 [W] Aufnahmegewicht eines Säuglings bis 12 Monate: Angabe fehlt

strcmp(MB->V0202,'A') == 0 && strlen(V4501) == 0 && strcmp(MB->V1203,'3') != 0 && MB->AGEJ < 1

B3168 [E] Das Aufnahmegewicht darf nicht kleiner als 0 oder grösser als 99999 Gramm sein

strlen(V4501) > 0 && (atol(V4501) < 0 || atol(V4501) > 16000)

B3169 [E] Das Aufnahmegewicht eines Säuglings muss - bei Eintrittsart = 3 Geburt - dem Geburtsgewicht im Neugeborenen-Datensatz entsprechen

MB->hasMN && strlen(V4501) > 0 && strcmp(MB->V1203,'3') == 0 && atol(V4501) != atol(MN->V2204)

B3170 [E] Körperlänge (cm): Angabe fehlt

strlen(V2205) == 0

B3171 [E] Körperlänge (cm): Ausserhalb des gültigen Wertebereichs

strlen(V2205) > 0 && strcmp(V2201,'1') == 0 && (strcmp(V2205,'0') == 0 || atol(V2205) > 65)

B3172 [W] Körperlänge (cm): Unwahrscheinlicher Wertebereich

strlen(V2205) > 0 && strcmp(V2201,'1') == 0 && (atol(V2205) < 25 && atol(V2205) > 0)

B3174 [W] Verhältnis Geburtsgewicht zu Körperlänge: Ausserhalb des gültigen Wertebereichs

atol(V2205) != 0 && (atol(V2204) / atol(V2205) < 20 || atol(V2204) / atol(V2205) > 120)

B3175 [W] Kopfumfang: Angabe fehlt

strcmp(MB->V0202,'A') == 0 && strlen(V4502) == 0 && MB->hasMN

B3176 [H] Kopfumfang: wird nur bei Neugeborenen erfasst.

strcmp(MB->V0202,'A') == 0 && strlen(V4502) > 0 && !MB->hasMN

ACTION: V4502=;

B3177 [W] Der Kopfumfang darf bei Lebendgeburten nicht kleiner als 15 oder grösser als 50 cm sein.

strlen(V4502) > 0 && strcmp(MB->V1502,'5') == 1 && (atol(V4502) < 15 || atol(V4502) > 50)

B3180 [E] Kongenitale Missbildungen: Angabe fehlt

strlen(V2206) == 0

B3181 [E] Kongenitale Missbildungen: Ungültige Angabe

strlen(V2206) > 0 && !isln(V2206,3,_3_KONGEN_MISSB)

B3190 [E] Verlegung Kind in anderes Spital: Angabe fehlt

strlen(V2207) == 0

B3191 [E] Verlegung Kind in anderes Spital: Ungültige Angabe

strlen(V2207) > 0 && !isln(V2207,2,_01)

B3195 [E] Geburtszeitpunkt: Angabe fehlt

strlen(V2103) == 0

B3196 [E] Geburtszeitpunkt: Ungültige Angabe

strlen(V2103) != 4 && (atoi(V2103)/100 > 23 || atoi(V2103)%100 > 59)

B4. Psychiatriedaten

B40. Patient

B4000 [W] Zivilstand: Angabe fehlt

strlen(V3201) == 0

ACTION: V3201="9";

B4001 [E] Zivilstand: Ungültige Angabe

strlen(V3201) > 0 && !isln(V3201,6,_6_ZIVILSTAND)

ACTION: V3201="9";

B4010 [W] Beschäftigung vor Eintritt (teilzeit erwerbstätig): Angabe fehlt

strlen(V3203) == 0

B4011 [E] Beschäftigung vor Eintritt (teilzeit erwerbstätig): Ungültige Angabe

strlen(V3203) > 0 && !isln(V3203,2,_01)

B4012 [W] Beschäftigung vor Eintritt (voll erwerbstätig): Angabe fehlt

strlen(V3204) == 0

B4013 [E] Beschäftigung vor Eintritt (voll erwerbstätig): Ungültige Angabe

strlen(V3204) > 0 && !isln(V3204,2,_01)

B4014 [W] Beschäftigung vor Eintritt (nicht erwerbstätig oder arbeitslos): Angabe fehlt

strlen(V3205) == 0

B4015 [E] Beschäftigung vor Eintritt (nicht erwerbstätig oder arbeitslos): Ungültige Angabe

strlen(V3205) > 0 && !isIn(V3205,2,_01)

B4016 [W] Beschäftigung vor Eintritt (Hausarbeit in eigenem Haushalt): Angabe fehlt

strlen(V3206) == 0

B4017 [E] Beschäftigung vor Eintritt (Hausarbeit in eigenem Haushalt): Ungültige Angabe

strlen(V3206) > 0 && !isIn(V3206,2,_01)

B4018 [W] Beschäftigung vor Eintritt (in Ausbildung): Angabe fehlt

strlen(V3207) == 0

B4019 [E] Beschäftigung vor Eintritt (in Ausbildung): Ungültige Angabe

strlen(V3207) > 0 && !isIn(V3207,2,_01)

B4020 [W] Beschäftigung vor Eintritt (Rehabilitationsprogramm): Angabe fehlt

strlen(V3208) == 0

B4021 [E] Beschäftigung vor Eintritt (Rehabilitationsprogramm): Ungültige Angabe

strlen(V3208) > 0 && !isIn(V3208,2,_01)

B4022 [W] Beschäftigung vor Eintritt (IV, AHV oder andere Rente): Angabe fehlt

strlen(V3209) == 0

B4023 [E] Beschäftigung vor Eintritt (IV, AHV oder andere Rente): Ungültige Angabe

strlen(V3209) > 0 && !isIn(V3209,2,_01)

B4024 [W] Beschäftigung vor Eintritt (Arbeit in geschütztem oder beschützendem Rahmen): Angabe fehlt

strlen(V3210) == 0

B4025 [E] Beschäftigung vor Eintritt (Arbeit in geschütztem oder beschützendem Rahmen): Ungültige Angabe

strlen(V3210) > 0 && !isIn(V3210,2,_01)

B4026 [W] Beschäftigung vor Eintritt (unbekannt): Angabe fehlt

strlen(V3211) == 0

B4027 [E] Beschäftigung vor Eintritt (unbekannt): Ungültige Angabe

strlen(V3211) > 0 && !isIn(V3211,2,_01)

B4040 [W] Höchste abgeschlossene Schul- oder Berufsbildung: Angabe fehlt

strlen(V3212) == 0

B4041 [E] Höchste abgeschlossene Schul- oder Berufsbildung: Ungültige Angabe

strlen(V3212) > 0 && !isln(V3212,7,_7_AUSBILD_PSY)

B41. Eintritt

B4110 [W] Aufenthaltsort vor Eintritt (Psychiatrie): Angabe fehlt

strlen(V3202) == 0

ACTION: V3202="90";

B4111 [E] Aufenthaltsort vor Eintritt (Psychiatrie): Ungültige Angabe

strlen(V3202) > 0 && !isln(V3202,22,_22_AUFEIN_PSY)

ACTION: V3202="90";

B4112 [W] Aufenthaltsort vor Eintritt (Psychiatrie): Detaillierte Angabe erforderlich (2-stellig)

strlen(V3202) > 0 && isln(V3202,9,_9_AUFEIN)

B4115 [E] Aufenthalt vor: Differenzen in Minimaldaten und Psychiatrie-Zusatz

strlen(V3202) > 0 && strcmp(MB->V0202,'A') == 0 && V3202[0] != MB->V1202[0]

B4130 [W] Einweisende Instanz (Psychiatrie): Angabe fehlt

strlen(V3301) == 0

ACTION: V3301="90";

B4131 [E] Einweisende Instanz (Psychiatrie): Ungültige Angabe

strlen(V3301) > 0 && !isln(V3301,20,_20_EININST_PSY)

ACTION: V3301="90";

B4132 [W] Einweisende Instanz (Psychiatrie): Detaillierte Angabe erforderlich (2-stellig)

strlen(V3301) > 0 && isln(V3301,8,_8_EININST)

ACTION: V3302="9";

B4135 [E] Einweisende Instanz: Differenzen in Minimaldaten und Psychiatrie-Zusatz

strlen(V3301) > 0 && strcmp(MB->V0202,'A') == 0 && V3301[0] != MB->V1204[0]

B4140 [W] Freiwilligkeit: Angabe fehlt

strlen(V3302) == 0

ACTION: V3302="9";

B4141 [E] Freiwilligkeit: Ungültige Angabe

strlen(V3302) > 0 && !isln(V3302,3,_3_FREIWILLIGKEIT)

B4142 [W] Die Angabe der Freiwilligkeit ist nicht mehr notwendig

strlen(V3302) > 0

B4150 [W] Fürsorgerische Unterbringung: Angabe fehlt

strlen(V3303) == 0

B4151 [E] Fürsorgerische Unterbringung: Ungültige Angabe

strlen(V3303) > 0 && !isln(V3303,2,_12)

B42. Spitalaufenthalt

B4210 [W] Anzahl Tage / Konsultationen: Angabe fehlt

strlen(V3401) == 0

B4211 [E] Anzahl Tage / Konsultationen: Ungültige Angabe

strlen(V3401) > 0 && (atol(V3401) < 0 || atol(V3401) > 9999)

B4220 [W] Behandlung (Psychiatrie): Angabe fehlt

strcmp(MB->V0202,'A') == 0 && strlen(V3402) == 0

B4221 [E] Behandlung (Psychiatrie): Ungültige Angabe

strlen(V3402) > 0 && !isln(V3402,12,_12_BEHPSY)

B4235 [H] Psychiatrie-Zusatzdaten / Hauptkostenstelle: Angabe wird in der Regel auf der Abteilung Psychiatrie (M500) erwartet

strcmp(MB->V1401,'M500') != 0

B43. Austritt

B4310 [W] Entscheid für Austritt (Psychiatrie): Angabe fehlt

strlen(V3501) == 0 && strcmp(MB->V0202,'A') == 0

ACTION: V3501="90";

B4311 [E] Entscheid für Austritt (Psychiatrie): Ungültige Angabe

strlen(V3501) > 0 && !isln(V3501,9,_9_AUSENT_PSY)

ACTION: V3501="90";

B4312 [W] Entscheid für Austritt (Psychiatrie): Detaillierte Angabe erforderlich (2-stellig)

strlen(V3501) > 0 && isln(V3501,7,_7_AUSENT)

B4315 [W] B-/C-Fälle mit MP: Austrittsmerkmale V3501 Entscheid für Austritt, V3502 Aufenthalt nach Austritt, V3503 Behandlung nach Austritt werden nicht ausgefüllt.

strcmp(MB->V0202,'A') != 0 && (strlen(V3501) > 0 || strlen(V3502) > 0 || strlen(V3503) > 0)

ACTION: V3501 = ; V3502=; V3503=;

B4316 [E] Entscheid für Austritt: Differenzen in Minimaldaten und Psychiatrie-Zusatz

strlen(V3501) > 0 && strcmp(MB->V0202,'A') == 0 && V3501[0] != MB->V1502[0]

B4320 [W] Aufenthalt nach Austritt (Psychiatrie): Angabe fehlt

strlen(V3502) == 0 && strcmp(MB->V0202,'A') == 0

ACTION: V3502="90";

B4321 [E] Aufenthalt nach Austritt (Psychiatrie): Angabe ungültig

strlen(V3502) > 0 && !isIn(V3502,13,_13_AUSAUF_PSY)

ACTION: V3502="90";

B4322 [W] Aufenthalt nach Austritt (Psychiatrie): Detaillierte Angabe erforderlich (2-stellig)

strlen(V3502) > 0 && isIn(V3502,10,_10_AUSAUF)

B4325 [E] Aufenthalt nach: Differenzen in Minimaldaten und Psychiatrie-Zusatz

strlen(V3502) > 0 && strcmp(MB->V0202,'A') == 0 && V3502[0] != MB->V1503[0]

B4330 [W] Behandlung nach Austritt (Psychiatrie): Angabe fehlt

strlen(V3503) == 0 && strcmp(MB->V0202,'A') == 0

ACTION: V3503="90";

B4331 [E] Behandlung nach Austritt (Psychiatrie): Ungültige Angabe

strlen(V3503) > 0 && !isIn(V3503,20,_20_AUSBEH_PSY)

ACTION: V3503="90";

B4332 [W] Behandlung nach Austritt (Psychiatrie): Detaillierte Angabe erforderlich (2-stellig)

strlen(V3503) > 0 && isIn(V3503,8,_8_AUSBEH)

B4335 [E] Behandlung nach: Differenzen in Minimaldaten und Psychiatrie-Zusatz

strlen(V3503) > 0 && strcmp(MB->V0202,'A') == 0 && V3503[0] != MB->V1504[0]

B47. Psychopharmakotherapie

B4700 [W] Psychopharmakotherapie: Neuroleptika: Angabe fehlt

strlen(V3403) == 0

B4701 [E] Psychopharmakotherapie: Neuroleptika: Ungültige Angabe

strlen(V3403) > 0 && !isIn(V3403,2,_01)

B4705 [W] Psychopharmakotherapie: Depotneuroleptika: Angabe fehlt

strlen(V3404) == 0

B4706 [E] Psychopharmakotherapie: Depotneuroleptika: Ungültige Angabe

strlen(V3404) > 0 && !isIn(V3404,2,_01)

B4710 [W] Psychopharmakotherapie: Antidepressiva: Angabe fehlt

strlen(V3405) == 0

B4711 [E] Psychopharmakotherapie: Antidepressiva: Ungültige Angabe

strlen(V3405) > 0 && !isIn(V3405,2,_01)

B4715 [W] Psychopharmakotherapie: Tranquilizer: Angabe fehlt

strlen(V3406) == 0

B4716 [E] Psychopharmakotherapie: Tranquilizer: Ungültige Angabe

strlen(V3406) > 0 && !isIn(V3406,2,_01)

B4720 [W] Psychopharmakotherapie: Hypnotika: Angabe fehlt

strlen(V3407) == 0

B4721 [E] Psychopharmakotherapie: Hypnotika: Ungültige Angabe

strlen(V3407) > 0 && !isIn(V3407,2,_01)

B4725 [W] Psychopharmakotherapie: Antiepileptika: Angabe fehlt

strlen(V3408) == 0

B4726 [E] Psychopharmakotherapie: Antiepileptika: Ungültige Angabe

strlen(V3408) > 0 && !isIn(V3408,2,_01)

B4730 [W] Psychopharmakotherapie: Lithium: Angabe fehlt

strlen(V3409) == 0

B4731 [E] Psychopharmakotherapie: Lithium: Ungültige Angabe

strlen(V3409) > 0 && !isIn(V3409,2,_01)

B4735 [W] Psychopharmakotherapie: Suchtsubstitutionsmittel: Angabe fehlt

strlen(V3410) == 0

B4736 [E] Psychopharmakotherapie: Suchtsubstitutionsmittel: Ungültige Angabe

strlen(V3410) > 0 && !isIn(V3410,2,_01)

B4740 [W] Psychopharmakotherapie: Suchtaversionsmittel: Angabe fehlt

strlen(V3411) == 0

B4741 [E] Psychopharmakotherapie: Suchtaversionsmittel: Ungültige Angabe

strlen(V3411) > 0 && !isIn(V3411,2,_01)

B4745 [W] Psychopharmakotherapie: Antiparkinsonmittel: Angabe fehlt

strlen(V3412) == 0

B4746 [E] Psychopharmakotherapie: Antiparkinsonmittel: Ungültige Angabe

strlen(V3412) > 0 && !isIn(V3412,2,_01)

B4750 [W] Psychopharmakotherapie: andere: Angabe fehlt

strlen(V3413) == 0

B4751 [E] Psychopharmakotherapie: andere: Ungültige Angabe

strlen(V3413) > 0 && !isIn(V3413,2,_01)

B4755 [W] Psychopharmakotherapie: Med. z. Beh. körperl. Leiden: Angabe fehlt

strlen(V3414) == 0

B4756 [E] Psychopharmakotherapie: Med. z. Beh. körperl. Leiden: Ungültige Angabe

strlen(V3414) > 0 && !isIn(V3414,2,_01)

B4900 [E] Ein stationärer A-Fall auf M500 Psychiatrie muss immer einen MP-Zusatzdatensatz enthalten

*MB->V0202 == 'A' && *MB->V1301 == '3' && strcmp(MB->V1401, 'M500') == 0 && !MB->hasMP

B5. MK

B50. MK CH

B5005 [E] Kantonskürzel im kantonalen Zusatzdatensatz: Angabe fehlt

isMKCH && strlen(V5102) == 0

B5006 [E] Kantonskürzel im kantonalen Zusatzdatensatz: Ungültige Angabe

isMKCH && strlen(V5102) > 0 && !isIn(V5102,26,_26_KANTON)

B51. MK GR

B5100 [E] Der obligatorische kantonale Zusatzdatensatz fehlt

strcmp(MB->V0202,'A') == 0 && MK->isMKGR && strcmp(V0304,'0') == 0

B5105 [W] 2. Feld (Kantonskürzel) im kantonalen Zusatzdatensatz GR: Angabe fehlt

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5102) == 0

B5106 [E] 2. Feld (Kantonskürzel) im kantonalen Zusatzdatensatz GR: Ungültige Angabe

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5102) > 0 && !isIn(V5102,26,_26_KANTON)

B5110 [W] 3. Feld (BUR-Nummer) im kantonalen Zusatzdatensatz GR: Angabe fehlt

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5103) == 0

B5111 [E] 3. Feld (BUR-Nummer) im kantonalen Zusatzdatensatz GR: Ungültige Angabe

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5103) > 0 && ((!isInt(V5103) || (atol(V5103) < 70000000 || atol(V5103) >= 90000000)))

B5115 [W] 4. Feld (Laufnummer) im kantonalen Zusatzdatensatz GR: Angabe fehlt

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5104) == 0

B5116 [E] 4. Feld (Laufnummer) im kantonalen Zusatzdatensatz GR: Ungültige Angabe

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5104) > 0 && strcmp(V5104,'1.0') != 0

B5120 [W] 5. Feld (PID) im kantonalen Zusatzdatensatz GR: Angabe fehlt

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5105) == 0

B5121 [E] 5. Feld (PID) im kantonalen Zusatzdatensatz GR: Ungültige Angabe

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5105) > 0 && !isAlphaNumUpperCase(V5105)

B5125 [W] 6. Feld (zu fakturierender DRG) im kantonalen Zusatzdatensatz GR: Angabe fehlt

strcmp(MB->V0202,'A') == 0 && !isIn(MB->V1401,3,_3_HKST_PSY_REHA) && isMKGR && strlen(V5106) == 0

B5130 [W] 7. Feld (Version DRG-Grouper) im kantonalen Zusatzdatensatz GR: Angabe fehlt

strcmp(MB->V0202,'A') == 0 && isMKGR && !isIn(MB->V1401,3,_3_HKST_PSY_REHA) && strlen(V5107) == 0

B5131 [E] 7. Feld (Version DRG-Grouper) im kantonalen Zusatzdatensatz GR: Ungültige Angabe

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5107) > 0 && !isIn(V5107,8,_8_SDRG_VERSION)

B5135 [W] 8. Feld (fakturiertes CW) im kantonalen Zusatzdatensatz GR: Angabe fehlt

strcmp(MB->V0202,'A') == 0 && isMKGR && !isIn(MB->V1401,3,_3_HKST_PSY_REHA) && strlen(V5108) == 0

B5136 [W] 8. Feld (fakturiertes CW) im kantonalen Zusatzdatensatz GR: Ungültige Angabe

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5108) > 0 && (!isFloat(V5108) || atof(V5108) < 0 || atof(V5108) > 9999)

B5140 [W] 9. Feld (Aufenthaltsdauer) im kantonalen Zusatzdatensatz GR: Angabe fehlt

strcmp(MB->V0202,'A') == 0 && isMKGR && !isIn(MB->V1401,3,_3_HKST_PSY_REHA) && strlen(V5109) == 0

B5141 [E] 9. Feld (Aufenthaltsdauer) im kantonalen Zusatzdatensatz GR: Ungültige Angabe

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5109) > 0 && (!isInt(V5109) || atol(V5109) < 0 || atol(V5109) > 9999)

B5145 [W] 10. Feld (BAG-Nummer Krankenkasse) im kantonalen Zusatzdatensatz GR: Angabe fehlt

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5110) == 0

B5146 [W] 10. Feld (BAG-Nummer Krankenkasse) im kantonalen Zusatzdatensatz GR: Ungültige Angabe

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5110) > 0 && (strcmp(V5110,'9999') == 0 || (!V5110_vsnr.inIndex && !isIn(V5110,5,_5_VSNR)))

B5150 [W] 11. Feld (Baserate) im kantonalen Zusatzdatensatz GR: Angabe fehlt

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5111) == 0

B5151 [E] 11. Feld (Baserate) im kantonalen Zusatzdatensatz GR: Ungültige Angabe

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5111) > 0 && (!isInt(V5111) || atol(V5111) < 0 || atol(V5111) > 99999)

B5155 [W] 12. Feld (Zusatzentgelte) im kantonalen Zusatzdatensatz GR: Angabe fehlt

strcmp(MB->V0202,'A') == 0 && isMKGR && !isIn(MB->V1401,3,_3_HKST_PSY_REHA) && strlen(V5112) == 0

B5156 [E] 12. Feld (Zusatzentgelte) im kantonalen Zusatzdatensatz GR: Ungültige Angabe

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5112) > 0 && (!isFloat(V5112) || atof(V5112) < 0 || atof(V5112) > 999999)

B5160 [W] 13. Feld (FID) im kantonalen Zusatzdatensatz GR: Angabe fehlt

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5113) == 0

B5161 [E] 13. Feld (FID) im kantonalen Zusatzdatensatz GR: Angabe darf max. 50 Zeichen lang sein

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5113) > 50

B5162 [E] 13. Feld (FID) im kantonalen Zusatzdatensatz GR: Ungültige Sonderzeichen

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5113) > 0 && !isAlphaNumUpperCase(V5113)

B5182 [W] 14. Feld (Reservefeld 2) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5114) > 0

B5183 [W] 15. Feld (Reservefeld 3) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5115) > 0

B5184 [W] 16. Feld (Reservefeld 4) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5116) > 0

B5185 [W] 17. Feld (Reservefeld 5) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5117) > 0

B5186 [W] 18. Feld (Reservefeld 6) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5118) > 0

B5187 [W] 19. Feld (Reservefeld 7) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5119) > 0

B5188 [W] 20. Feld (Reservefeld 8) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5120) > 0

B5189 [W] 21. Feld (Reservefeld 9) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5121) > 0

B5190 [W] 22. Feld (Reservefeld 10) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5122) > 0

B5191 [W] 23. Feld (Reservefeld 11) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5123) > 0

B5192 [W] 24. Feld (Reservefeld 12) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5124) > 0

B5193 [W] 25. Feld (Reservefeld 13) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5125) > 0

B5194 [W] 26. Feld (Reservefeld 14) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5126) > 0

B5195 [W] 27. Feld (Reservefeld 15) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

```
strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5127) > 0
```

B5196 [W] 28. Feld (Reservefeld 16) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

```
strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5128) > 0
```

B5197 [W] 29. Feld (Reservefeld 17) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

```
strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5129) > 0
```

B5198 [W] 30. Feld (Reservefeld 18) im kantonalen Zusatzdatensatz GR: enthält Werte, obwohl das Feld noch nicht definiert ist.

```
strcmp(MB->V0202,'A') == 0 && isMKGR && strlen(V5130) > 0
```

B52. MK LU

B5205 [E] 2. Feld (Kantonskürzel) im kantonalen Zusatzdatensatz: Angabe fehlt

```
isMKLU && strlen(V5102) == 0
```

B5206 [E] Kantonskürzel im kantonalen Zusatzdatensatz: Ungültige Angabe

```
isMKLU && strlen(V5102) > 0 && !isIn(V5102,6,_6_KTLU)
```

B5207 [E] Kantonskürzel im MB-Record: Ungültige Angabe

```
isMKLU && strlen(MB->V0104) > 0 && !isIn(MB->V0104,6,_6_KTLU)
```

B5208 [E] Unterschiedliche Kantonskürzel im MB-/MK-Record: Ungültige Angabe

```
isMKLU && strlen(MB->V0104) > 0 && strlen(V5102) > 0 && strcmp(MB->V0104,V5102) != 0
```

B5210 [E] 3. Feld (Burnummer) im kantonalen Zusatzdatensatz: Angabe fehlt

```
isMKLU && strlen(V5103) == 0
```

B5215 [E] 4. Feld (Patientenfallnummer) im kantonalen Zusatzdatensatz: Angabe fehlt

```
isMKLU && strlen(V5104) == 0
```

B53. MK BE Psy

B5302 [X] mkBePsy Wohnform: Angabe fehlt

```
strlen(MK->V6202) == 0
```

B5303 [X] mkBePsy Anzahl Behandlungen: Angabe fehlt

```
strlen(MK->V6203) == 0
```

B5304 [X] mkBePsy Erster Kontakt mit PSY: Angabe fehlt

strlen(MK->V6204) == 0

B5305 [X] mkBePsy Umfang des Hilfesystems: Angabe fehlt

strlen(MK->V6205) == 0

B5306 [X] mkBePsy Schul. Unterstützungsbdarf: Angabe fehlt

strlen(MK->V6206) == 0

B5307 [X] mkBePsy GAF bei Eintritt: Angabe fehlt

strlen(MK->V6207) == 0

B5350 [X] mkBePsy Angebotsart: Angabe fehlt

strlen(MK->VLM03) == 0

B6. Rehospitalisationsangaben

B6640 [E] Weitere Wiedereintritte: Ungültige Angabe

strlen(V4741) > 0 && !(strcmp(V4741,'0') == 0 || strcmp(V4741,'1') == 0)

B6900 [E] 1. Zwischenaustritt: Ungültige Datumsangabe

!dateV4701.isNull && !dateV4701.isYYYYMMDD

B6901 [E] 1. Wiedereintritt: Ungültige Datumsangabe

!dateV4702.isNull && !dateV4702.isYYYYMMDD

B6902 [E] 2. Zwischenaustritt: Ungültige Datumsangabe

!dateV4711.isNull && !dateV4711.isYYYYMMDD

B6903 [E] 2. Wiedereintritt: Ungültige Datumsangabe

!dateV4712.isNull && !dateV4712.isYYYYMMDD

B6904 [E] 3. Zwischenaustritt: Ungültige Datumsangabe

!dateV4721.isNull && !dateV4721.isYYYYMMDD

B6905 [E] 3. Wiedereintritt: Ungültige Datumsangabe

!dateV4722.isNull && !dateV4722.isYYYYMMDD

B6906 [E] 4. Zwischenaustritt: Ungültige Datumsangabe

!dateV4731.isNull && !dateV4731.isYYYYMMDD

B6907 [E] 4. Wiedereintritt: Ungültige Datumsangabe

!dateV4732.isNull && !dateV4732.isYYYYMMDD

B6910 [E] Der 1. Zwischenaustritt (4.7.V01) liegt vor dem Eintrittsdatum

strcmp(MB->V0202,'A') == 0 && !MB->dateV1201.isNull && !dateV4701.isNull && MB->dateV1201.DateValue > dateV4701.DateValue

B6911 [E] Der 1. Wiedereintritt (4.7.V02) liegt vor dem Eintrittsdatum

strcmp(MB->V0202,'A') == 0 && !MB->dateV1201.isNull && !dateV4702.isNull && MB->dateV1201.DateValue > dateV4702.DateValue

B6912 [E] Der 2. Zwischenaustritt (4.7.V11) liegt vor dem Eintrittsdatum

strcmp(MB->V0202,'A') == 0 && !MB->dateV1201.isNull && !dateV4711.isNull && MB->dateV1201.DateValue > dateV4711.DateValue

B6913 [E] Der 2. Wiedereintritt (4.7.V12) liegt vor dem Eintrittsdatum

strcmp(MB->V0202,'A') == 0 && !MB->dateV1201.isNull && !dateV4712.isNull && MB->dateV1201.DateValue > dateV4712.DateValue

B6914 [E] Der 3. Zwischenaustritt (4.7.V21) liegt vor dem Eintrittsdatum

strcmp(MB->V0202,'A') == 0 && !MB->dateV1201.isNull && !dateV4721.isNull && MB->dateV1201.DateValue > dateV4721.DateValue

B6915 [E] Der 3. Wiedereintritt (4.7.V22) liegt vor dem Eintrittsdatum

strcmp(MB->V0202,'A') == 0 && !MB->dateV1201.isNull && !dateV4722.isNull && MB->dateV1201.DateValue > dateV4722.DateValue

B6916 [E] Der 4. Zwischenaustritt (4.7.V31) liegt vor dem Eintrittsdatum

strcmp(MB->V0202,'A') == 0 && !MB->dateV1201.isNull && !dateV4731.isNull && MB->dateV1201.DateValue > dateV4731.DateValue

B6917 [E] Der 4. Wiedereintritt (4.7.V32) liegt vor dem Eintrittsdatum

strcmp(MB->V0202,'A') == 0 && !MB->dateV1201.isNull && !dateV4732.isNull && MB->dateV1201.DateValue > dateV4732.DateValue

B6920 [E] Der 1. Zwischenaustritt (4.7.V01) liegt nach dem Austrittsdatum

strcmp(MB->V0202,'A') == 0 && !MB->dateV1501.isNull && !dateV4701.isNull && dateV4701.DateValue > MB->dateV1501.DateValue

B6921 [E] Der 1. Wiedereintritt (4.7.V02) liegt nach dem Austrittsdatum

strcmp(MB->V0202,'A') == 0 && !MB->dateV1501.isNull && !dateV4702.isNull && dateV4702.DateValue > MB->dateV1501.DateValue

B6922 [E] Der 2. Zwischenaustritt (4.7.V11) liegt nach dem Austrittsdatum

strcmp(MB->V0202,'A') == 0 && !MB->dateV1501.isNull && !dateV4711.isNull && dateV4711.DateValue > MB->dateV1501.DateValue

B6923 [E] Der 2. Wiedereintritt (4.7.V12) liegt nach dem Austrittsdatum

strcmp(MB->V0202,'A') == 0 && !MB->dateV1501.isNull && !dateV4712.isNull && dateV4712.DateValue > MB->dateV1501.DateValue

B6924 [E] Der 3. Zwischenaustritt (4.7.V21) liegt nach dem Austrittsdatum

strcmp(MB->V0202,'A') == 0 && !MB->dateV1501.isNull && !dateV4721.isNull && dateV4721.DateValue > MB->dateV1501.DateValue

B6925 [E] Der 3. Wiedereintritt (4.7.V22) liegt nach dem Austrittsdatum

strcmp(MB->V0202,'A') == 0 && !MB->dateV1501.isNull && !dateV4722.isNull && dateV4722.DateValue > MB->dateV1501.DateValue

B6926 [E] Der 4. Zwischenaustritt (4.7.V31) liegt nach dem Austrittsdatum

strcmp(MB->V0202,'A') == 0 && !MB->dateV1501.isNull && !dateV4731.isNull && dateV4731.DateValue > MB->dateV1501.DateValue

B6927 [E] Der 4. Wiedereintritt (4.7.V32) liegt nach dem Austrittsdatum

strcmp(MB->V0202,'A') == 0 && !MB->dateV1501.isNull && !dateV4732.isNull && dateV4732.DateValue > MB->dateV1501.DateValue

B6930 [E] Der erste Zwischenaustritt muss vor dem zweiten Zwischenaustritt liegen

strcmp(MB->V0202,'A') == 0 && !dateV4701.isNull && !dateV4711.isNull && dateV4701.DateValue > dateV4711.DateValue

B6931 [E] Der zweite Zwischenaustritt muss vor dem dritten Zwischenaustritt liegen

strcmp(MB->V0202,'A') == 0 && !dateV4711.isNull && !dateV4721.isNull && dateV4711.DateValue > dateV4721.DateValue

B6932 [E] Der dritte Zwischenaustritt muss vor dem vierten Zwischenaustritt liegen

strcmp(MB->V0202,'A') == 0 && !dateV4721.isNull && !dateV4731.isNull && dateV4721.DateValue > dateV4731.DateValue

B6940 [E] Der erste Wiedereintritt muss vor dem zweiten Wiedereintritt liegen

strcmp(MB->V0202,'A') == 0 && !dateV4702.isNull && !dateV4712.isNull && dateV4702.DateValue > dateV4712.DateValue

B6941 [E] Der zweite Wiedereintritt muss vor dem dritten Wiedereintritt liegen

strcmp(MB->V0202,'A') == 0 && !dateV4712.isNull && !dateV4722.isNull && dateV4712.DateValue > dateV4722.DateValue

B6942 [E] Der dritte Wiedereintritt muss vor dem vierten Wiedereintritt liegen

strcmp(MB->V0202,'A') == 0 && !dateV4722.isNull && !dateV4732.isNull && dateV4722.DateValue > dateV4732.DateValue

B6950 [E] Grund des 1. Wiedereintrittes: Ungültige Angabe

strlen(V4703) > 0 && !isIn(V4703,4,_1239)

B6951 [E] Grund des 2. Wiedereintrittes: Ungültige Angabe

strlen(V4713) > 0 && !isIn(V4713,4,_1239)

B6952 [E] Grund des 3. Wiedereintrittes: Ungültige Angabe

strlen(V4723) > 0 && !isIn(V4723,4,_1239)

B6953 [E] Grund des 4. Wiedereintrittes: Ungültige Angabe

strlen(V4733) > 0 && !isIn(V4733,4,_1239)

B6960 [E] Wenn ein 1. Zwischenaustritt angegeben wird, muss der 1. Wiedereintritt ausgefüllt sein

strlen(V4701) > 0 && strlen(V4702) == 0

B6961 [E] Wenn ein 2. Zwischenaustritt angegeben wird, muss der 2. Wiedereintritt ausgefüllt sein

strlen(V4711) > 0 && strlen(V4712) == 0

B6962 [E] Wenn ein 3. Zwischenaustritt angegeben wird, muss der 3. Wiedereintritt ausgefüllt sein

strlen(V4721) > 0 && strlen(V4722) == 0

B6963 [E] Wenn ein 4. Zwischenaustritt angegeben wird, muss der 4. Wiedereintritt ausgefüllt sein

strlen(V4731) > 0 && strlen(V4732) == 0

B6970 [E] Wenn ein 1. Wiedereintritt angegeben wird, muss auch dessen Grund ausgefüllt sein

strlen(V4702) > 0 && strlen(V4703) == 0

B6971 [E] Wenn ein 2. Wiedereintritt angegeben wird, muss auch dessen Grund ausgefüllt sein

strlen(V4712) > 0 && strlen(V4713) == 0

B6972 [E] Wenn ein 3. Wiedereintritt angegeben wird, muss auch dessen Grund ausgefüllt sein

strlen(V4722) > 0 && strlen(V4723) == 0

B6973 [E] Wenn ein 4. Wiedereintritt angegeben wird, muss auch dessen Grund ausgefüllt sein

strlen(V4732) > 0 && strlen(V4733) == 0

B6975 [E] Der Fall hat einen 1. Zwischenaustritt im Vorjahr, einen Austritt im laufenden Jahr. Diese Rehosp-Erfassung entspricht nicht den SwissDRG-Regeln.

strcmp(MB->V0202,'A') == 0 && dateV4701.isYYYYMMDD && dateV4701.Year != MB->dateV1501.Year

B6976 [E] Die Dauer des administrativen Urlaubs entspricht der Länge der Abwesenheiten zwischen den Aus-/Wiedereintritten. Wurde die Abwesenheit doppelt erfasst?

dateV4701.hasDate && UrlaubT > 0 && UrlaubT == ABS

B6977 [E] Die berechnete Aufenthaltsdauer beträgt weniger als 1 Tag.

strcmp(MB->V0202,'A') == 0 && LOS < 1

B6978 [E] Ein Wiedereintritt liegt mehr als 18 Tage hinter dem ersten Zwischenaustritt. Bitte prüfen Sie den Rehosp-Verlauf dieses Falles.

(dateV4702.hasDate && dateV4702.DateValue - dateV4701.DateValue > 18) || (dateV4712.hasDate && dateV4712.DateValue - dateV4701.DateValue > 18) || (dateV4722.hasDate && dateV4722.DateValue - dateV4701.DateValue > 18) || (dateV4732.hasDate && dateV4732.DateValue - dateV4701.DateValue > 18)

B6980 [E] Der 1. Wiedereintritt liegt mehr als 18 Tage hinter dem 1. Zwischenaustritt.

dateV4701.isYYYYMMDD && dateV4702.isYYYYMMDD && dateV4702.DateValue - dateV4701.DateValue > 18

B6981 [E] Der 2. Wiedereintritt liegt mehr als 18 Tage hinter dem 1. Zwischenaustritt.

dateV4701.isYYYYMMDD && dateV4712.isYYYYMMDD && dateV4712.DateValue - dateV4701.DateValue > 18

B6982 [E] Der 3. Wiedereintritt liegt mehr als 18 Tage hinter dem 1. Zwischenaustritt.

dateV4701.isYYYYMMDD && dateV4722.isYYYYMMDD && dateV4722.DateValue - dateV4701.DateValue > 18

B6983 [E] Der 4. Wiedereintritt liegt mehr als 18 Tage hinter dem 1. Zwischenaustritt.

dateV4701.isYYYYMMDD && dateV4732.isYYYYMMDD && dateV4732.DateValue - dateV4701.DateValue > 18

B6984 [E] 1. Zwischenaustritt/1. Wiedereintritt: unplausible Angaben.

strcmp(MB->V0202,'A') == 0 && (strlen(V4701) > 0 || strlen(V4702) > 0) && (!dateV4701.isYYYYMMDD || !dateV4702.isYYYYMMDD || dateV4701.DateValue < MB->dateV1201.DateValue || dateV4702.DateValue < dateV4701.DateValue || dateV4702.DateValue > MB->dateV1501.DateValue)

B6985 [E] 2. Zwischenaustritt/2. Wiedereintritt: unplausible Angaben.

strcmp(MB->V0202,'A') == 0 && (strlen(V4711) > 0 || strlen(V4712) > 0) && (!dateV4711.isYYYYMMDD || !dateV4712.isYYYYMMDD || dateV4711.DateValue < dateV4702.DateValue || dateV4712.DateValue < dateV4711.DateValue || dateV4712.DateValue > MB->dateV1501.DateValue)

B6986 [E] 3. Zwischenaustritt/3. Wiedereintritt: unplausible Angaben.

strcmp(MB->V0202,'A') == 0 && (strlen(V4721) > 0 || strlen(V4722) > 0) && (!dateV4721.isYYYYMMDD || !dateV4722.isYYYYMMDD || dateV4721.DateValue < dateV4712.DateValue || dateV4722.DateValue < dateV4721.DateValue || dateV4722.DateValue > MB->dateV1501.DateValue)

B6987 [E] 4. Zwischenaustritt/4. Wiedereintritt: unplausible Angaben.

strcmp(MB->V0202,'A') == 0 && (strlen(V4731) > 0 || strlen(V4732) > 0) && (!dateV4731.isYYYYMMDD || !dateV4732.isYYYYMMDD || dateV4731.DateValue < dateV4722.DateValue || dateV4732.DateValue < dateV4731.DateValue || dateV4732.DateValue > MB->dateV1501.DateValue)

C. Formelle Fehler

C0. Strukturelle Probleme

C0000 [E] Anzahl gelieferte Datensätze: Angabe in Lieferungsmeldung überschritten

LineNumberInMX > atoi(MX->V0008)

C0220 [E] MB: Statistikfall A-Test: Austritt muss in Erhebungsperiode liegen

strcmp(V0202,'A') == 0 && dateV1501.hasDate && (dateV1501.DateValue < MX->dateV0006.DateValue || dateV1501.DateValue > MX->dateV0007.DateValue)

ACTION: doWriteRec=FALSE;

C0301 [E] Neugeborenen-Zusatzdaten: Zusatzdatensatz vorhanden ohne Meldung im Basisdatensatz

strcmp(MB->V0301,'0') == 0

ACTION: MB->V0301="1";

C0320 [E] MN: Statistikfall A-Test: Austritt muss in Erhebungsperiode liegen

strcmp(MB->V0202,'A') == 0 && MB->dateV1501.hasDate && (MB->dateV1501.DateValue < MX->dateV0006.DateValue || MB->dateV1501.DateValue > MX->dateV0007.DateValue)

C0400 [E] Psychiatrie-Zusatzdaten: Fehlender Zusatzdatensatz

strcmp(V0302,'1') == 0 && !hasMP

ACTION: V0302="0";

C0401 [E] Psychiatrie-Zusatzdaten: Zusatzdatensatz vorhanden ohne Meldung im Basisdatensatz

strcmp(MB->V0302,'0') == 0

C0420 [E] MP: Statistikfall A-Test: Austritt muss in Erhebungsperiode liegen

strcmp(MB->V0202,'A') == 0 && MB->dateV1501.hasDate && (MB->dateV1501.DateValue < MX->dateV0006.DateValue || MB->dateV1501.DateValue > MX->dateV0007.DateValue)

ACTION: MB->V0302="0"; MB->hasMP=FALSE;

C0600 [E] Kantonale Zusatzdaten: Fehlender Zusatzdatensatz

strcmp(V0304,'1') == 0 && !hasMK

ACTION: V0304="0";

C0601 [E] Kantonale Zusatzdaten: Zusatzdatensatz vorhanden ohne Meldung im Basisdatensatz

strcmp(MB->V0304,'0') == 0

ACTION: MB->V0304="1";

C0620 [E] MK: Statistikfall A-Test: Austritt muss in Erhebungsperiode liegen

strcmp(MB->V0202,'A') == 0 && MB->dateV1501.hasDate && (MB->dateV1501.DateValue < MX->dateV0006.DateValue || MB->dateV1501.DateValue > MX->dateV0007.DateValue)

C0700 [E] Patientengruppen-Zusatzdaten: Fehlender Zusatzdatensatz

strcmp(V0303,'1') == 0 && !hasMD

ACTION: V0303="0";

C0701 [E] Patientengruppen-Zusatzdaten: Zusatzdatensatz vorhanden ohne Meldung im Basisdatensatz

strcmp(MB->V0303,'0') == 0

ACTION: MB->V0303="1";

C0710 [E] Patientengruppen-Zusatzdaten: Fehlender obligatorischer MD-Zusatzdatensatz

strcmp(V0202,'A') == 0 && strcmp(V1301,'3') == 0 && !hasMD

C0720 [E] MD: Statistikfall A-Test: Austritt muss in Erhebungsperiode liegen

strcmp(MB->V0202,'A') == 0 && MB->dateV1501.hasDate && (MB->dateV1501.DateValue < MX->dateV0006.DateValue || MB->dateV1501.DateValue > MX->dateV0007.DateValue)

C0900 [E] Nicht erlaubtes Sonderzeichen im MX

checkChars() != -1

C0901 [E] Nicht erlaubtes Sonderzeichen im MB

checkChars() != -1

C0902 [E] Nicht erlaubtes Sonderzeichen im MN

checkChars() != -1

C0903 [E] Nicht erlaubtes Sonderzeichen im MP

checkChars() != -1

C0904 [E] Nicht erlaubtes Sonderzeichen im MD

checkChars() != -1

C0905 [E] Nicht erlaubtes Sonderzeichen im MK

checkChars() != -1

C1. MX-Probleme

C10. MX Missing

C1010 [E] BUR-Nummer: Angabe fehlt

strlen(V0002) == 0

C1020 [E] Kürzel der Klinik: Angabe fehlt

strlen(V0003) == 0

ACTION: V0003="Dummy";

C1021 [E] Kürzel der Klinik: Angabe darf max. 8 Zeichen lang sein

strlen(V0003) > 8

ACTION: V0003[8]=0;

C1030 [E] Export-Erstellungsdatum: Angabe fehlt

dateV0004.isNull

C1031 [E] Export-Erstellungsdatum: Ungültige Datumsangabe

!(dateV0004.isNull || dateV0004.isYYYYMMDD)

ACTION: if (strlen(V0004) > 8) V0004[8]=0;

C1040 [E] Laufnummer Datenmeldung: Angabe fehlt

strlen(V0005) == 0

ACTION: V0005="1";

C1041 [E] Laufnummer Datenmeldung: Ungültige Angabe

strlen(V0005) > 0 && ((!isInt(V0005) || atol(V0005) < 1 || atol(V0005) > 99999))

ACTION: V0005="1";

C1050 [E] Meldungsperiode von: Angabe fehlt

dateV0006.isNull

C1051 [E] Meldungsperiode von: Ungültige Datumsangabe

!(dateV0006.isNull || dateV0006.isYYYYMMDD)

ACTION: if (strlen(V0006) > 8) V0006[8]=0;

C1060 [E] Meldungsperiode bis: Angabe fehlt

dateV0007.isNull

C1061 [E] Meldungsperiode bis: Ungültige Datumsangabe

!(dateV0007.isNull || dateV0007.isYYYYMMDD)

ACTION: if (strlen(V0007) > 8) V0007[8]=0;

C1070 [E] Anzahl Datensätze: Angabe fehlt

strlen(V0008) == 0

C1071 [E] Anzahl Datensätze: Ungültige Angabe

strlen(V0008) != 0 && ((!isInt(V0008) || atol(V0008) < 3 || atol(V0008) > 99999999))

C1080 [E] C-Schlüssel Feld: Angabe fehlt

strlen(V0009) == 0

C1081 [E] C-Schlüssel Feld: Falscher Datentyp (nicht hexadezimal)

strlen(V0009) != 0 && !isHex(V0009)

C1082 [E] C-Schlüssel Feld: Länge nicht korrekt

strlen(V0009) > 0 && strlen(V0009) != 256

C1091 [E] Lieferungsmeldung Inkonsistenz zwischen Meldeperiode von, bis und Export-Erstellungsdatum

dateV0007.DateValue < dateV0006.DateValue || dateV0004.DateValue < dateV0007.DateValue

C1210 [E] Pseudo Record "Meier Hans": Nur die Rekordart und der Verbindungscode dürfen ausgefüllt sein

strlen(V0102) > 0 || strlen(V0103) > 0 || strlen(V0104) > 0 || strlen(V0202) > 0 || strlen(V0301) > 0 || strlen(V0302) > 0 || strlen(V0303) > 0 || strlen(V0304) > 0 || strlen(V1101) > 0 || strlen(V1102) > 0 || strlen(V1103) > 0 || strlen(V1104) > 0 || strlen(V1105) > 0 || strlen(V1201) > 0 || strlen(V1202) > 0 || strlen(V1203) > 0 || strlen(V1204) > 0 || strlen(V1301) > 0 || strlen(V1302) > 0 || strlen(V1303) > 0 || strlen(V1304) > 0 || strlen(V1401) > 0 || strlen(V1402) > 0 || strlen(V1501) > 0 || strlen(V1502) > 0 || strlen(V1503) > 0 || strlen(V1504) > 0 || strlen(DIA[0].Key) > 0

C1240 [E] Pseudo Record "Meier Hans": Verbindungscode im Testrecord fehlt

strlen(V0201) == 0

C1241 [E] Pseudo Record "Meier Hans": Verbindungscode im Testrecord ist nicht hexadezimal

strlen(V0201) > 0 && !isHex(V0201)

C1242 [E] Anonymer Verbindungskode: Angabe ungültig, beruht auf fehlenden oder ungültigen Werten in Name, Vorname, Geburtsdatum (vollständig) oder Geschlecht vor dem Anonymisieren

strcmp(V0201,'0000000000000000') == 0

C1243 [E] Pseudo Record "Meier Hans": Die Zeichenlänge des Verbindungscodes ist nicht korrekt

strlen(V0201) > 0 && strlen(V0201) != 16

C2. MB

C2010 [E] BUR-Nummer: Angabe fehlt

strlen(V0102) == 0

C2015 [E] BUR-Nummer: Widerspruch mit der Angabe in der Lieferungsmeldung

strlen(V0102) > 0 && strcmp(V0102,MX->V0002) != 0

C2020 [E] Standort: Angabe fehlt

strlen(V0103) == 0

C2021 [E] Standort: Ungültige Angabe

strlen(V0103) > 0 && !isIn (V0103,36,_NOGA)

ACTION: V0103=;

C2022 [E] Standort: Angabe darf max. 2 Zeichen lang sein

strlen(V0103) > 2

C2030 [E] Kanton: Angabe fehlt

strlen(V0104) == 0

C2031 [E] Kanton: Ungültige Angabe

strlen(V0104) > 0 && !isIn (V0104,26,_26_KANTON)

C2040 [E] Anonymer Verbindungscode: Angabe fehlt

strlen(V0201) == 0

C2041 [E] Anonymer Verbindungscode: Zeichenlänge nicht korrekt

strlen(V0201) > 0 && strlen(V0201) != 16

C2042 [E] Anonymer Verbindungscode: Angabe ist nicht hexadezimal

strlen(V0201) > 0 && !isHex(V0201)

C2045 [E] Anonymer Verbindungskode: Angabe ungültig, beruht auf fehlenden oder ungültigen Werten in Name, Vorname, Geburtsdatum (vollständig) oder Geschlecht vor dem Anonymisieren

strcmp(V0201,'0000000000000000') == 0

C2050 [E] Kennzeichnung Statistikfall: Angabe fehlt

strlen(V0202) == 0

C2051 [E] Kennzeichnung Statistikfall: Ungültige Angabe

strlen(V0202) > 0 && !isIn(V0202,3,_ABC)

C2060 [E] Neugeborenen-Zusatzdaten: Angabe fehlt

strlen(V0301) == 0

ACTION: V0301="0";

C2061 [E] Neugeborenen-Zusatzdaten: Ungültige Angabe

strlen(V0301) > 0 && !isIn(V0301,2,_01)

ACTION: V0301="0";

C2070 [E] Psychiatrie-Zusatzdaten: Angabe fehlt

strlen(V0302) == 0

ACTION: V0302="0";

C2071 [E] Psychiatrie-Zusatzdaten: Ungültige Angabe

strlen(V0302) > 0 && !isIn(V0302,2,_01)

ACTION: V0302="0";

C2080 [E] Patientengruppen-Zusatzdaten: Angabe fehlt

strlen(V0303) == 0

ACTION: V0303="0";

C2081 [E] Patientengruppen-Zusatzdaten: Ungültige Angabe

strlen(V0303) > 0 && !isIn(V0303,2,_01)

ACTION: V0303="0";

C2090 [E] Kantonale Zusatzdaten: Angabe fehlt

strlen(V0304) == 0

ACTION: V0304="0";

C2091 [E] Kantonale Zusatzdaten: Ungültige Angabe

strlen(V0304) > 0 && !isIn(V0304,2,_01)

ACTION: V0304="0";

C2111 [H] Geburtsdatum: Die Länge beträgt 8 Zeichen

!dateV1102.isNull && strlen(V1102) != 8

ACTION: if (strlen(V1102) > 8) V1102[8]=0;

C2210 [W] Die Diagnose im MB muss der Diagnose im MD entsprechen

strcmp(MB->V0202,'A') == 0 && strcmp(MB->DIA[i].Key,DIA[i].Key) != 0

C2220 [W] Die Behandlung im MB muss der Behandlung im MD entsprechen

strcmp(MB->V0202,'A') == 0 && strncmp(BEH[i].Key,MB->BEH[i].Key,5) != 0 && (!MB->BEH[i].inIndex || strncmp(MB->BEH[i].Key,BEH[i].Key,strlen(MB->BEH[i].Key)))

C2222 [E] Beginn der Hauptbehandlung MD muss dem Beginn der Hauptbehandlung MB entsprechen

strcmp(MB->V0202,'A') == 0 && strcmp(BEH[0].dateDatum.Value,MB->V1702) != 0

C2320 [E] Statistikfall B-Test: Eintritt muss in Erhebungsperiode liegen

V0202[0] == 'B' && dateV1201.hasDate && (dateV1201.DateValue < MX->dateV0006.DateValue || dateV1201.DateValue > MX->dateV0007.DateValue)

C2321 [E] Statistikfall B-Test: Austritt kann nicht vor Ende der Meldeperiode sein

strcmp(V0202,'B') == 0 && dateV1501.hasDate && dateV1501.DateValue < MX->dateV0007.DateValue

C2322 [H] Statistikfall B-Test: Angaben werden nur bis 1.4.V02 (Hauptkostenträger) erwartet

V0202[0] == 'B' && (strlen(V1501) > 0 || strlen(V1502) > 0 || strlen(V1503) > 0 || strlen(V1504) > 0 || DIACount > 0 || BEHCount > 0)

ACTION: V1501=; V1502=; V1503=; V1504=; V1702=; for (int di=0; di<=DIACount;di++) DIA[di].Key=; for (int bi=0; bi<=BEHCount;bi++) BEH[bi].Key=;

C2330 [E] Statistikfall C-Test: Eintritt muss vor Erhebungsperiode liegen

strcmp(V0202,'C') == 0 && dateV1201.hasDate && dateV1201.DateValue > MX->dateV0006.DateValue

C2331 [E] Statistikfall C-Test: Austritt kann nicht vor Ende der Meldeperiode sein

V0202[0] == 'C' && dateV1501.hasDate && dateV1501.DateValue < MX->dateV0007.DateValue

C2332 [H] Statistikfall C-Test: Austrittsangaben werden nicht erwartet

strcmp(V0202,'C') == 0 && (strlen(V1501) > 0 || strlen(V1502) > 0 || strlen(V1503) > 0 || strlen(V1504) > 0)

ACTION: V1501=; V1502=; V1503=; V1504=; V1702=;

C7. MD

C7010 [E] Lokalisation des Spitals: Angabe fehlt

strlen(V4102) == 0

C7011 [W] Lokalisation des Spitals: ungültige Angabe (gültige PLZ erforderlich)

!V4102_plz.isNull && !V4102_plz.inIndex

C7029 [E] Das Reservefeld zur Lokalisation des Spitals ist noch nicht definiert. Es muss leer sein.

strlen(V4103) > 0

C7051 [E] Fallnummer (Fallkostenstatistik): Angabe darf max. 16 Zeichen lang sein

strlen(V4601) > 16

C7052 [E] Fallnummer (Fallkostenstatistik): Ungültige Sonderzeichen

strlen(V4601) > 0 && ! isAlphaNumUpperCase(V4601)

C7061 [W] DRG-Status: Angabe darf max. 1 Zeichen lang sein

strlen(V4801) > 1

C7100 [E] Die Diagnosecodes müssen nacheinander aufgelistet werden. Es dürfen innerhalb der übermittelten Codes keine leeren ICD-Felder ('Lücken') enthalten sein.

i >0 && !DIA[i].isNull && ((i<3 && DIA[0].isNull) || (i>2 && DIA[i-1].isNull))

C7200 [E] Die Behandlungscodes müssen nacheinander aufgelistet werden. Es dürfen innerhalb der übermittelten Codes keine leeren CHOP-Felder ('Lücken') enthalten sein.

i >0 && !BEH[i].isNull && BEH[i-1].isNull

C7301 [W] Medikament 1: Angabe darf max. 50 Zeichen lang sein

strlen(V4802) > 50

C7302 [W] Medikament 2: Angabe darf max. 50 Zeichen lang sein

strlen(V4803) > 50

C7303 [W] Medikament 3: Angabe darf max. 50 Zeichen lang sein

strlen(V4804) > 50

C7304 [W] Medikament 4: Angabe darf max. 50 Zeichen lang sein

strlen(V4805) > 50

C7305 [W] Medikament 5: Angabe darf max. 50 Zeichen lang sein

strlen(V4806) > 50

C7306 [W] Medikament 6: Angabe darf max. 50 Zeichen lang sein

strlen(V4807) > 50

C7307 [W] Medikament 7: Angabe darf max. 50 Zeichen lang sein

strlen(V4808) > 50

C7308 [W] Medikament 8: Angabe darf max. 50 Zeichen lang sein

strlen(V4809) > 50

C7309 [W] Medikament 9: Angabe darf max. 50 Zeichen lang sein

strlen(V4810) > 50

C7310 [W] Medikament 10: Angabe darf max. 50 Zeichen lang sein

strlen(V4811) > 50

C7311 [W] Medikament 11: Angabe darf max. 50 Zeichen lang sein

strlen(V4812) > 50

C7312 [W] Medikament 12: Angabe darf max. 50 Zeichen lang sein

strlen(V4813) > 50

C7313 [W] Medikament 13: Angabe darf max. 50 Zeichen lang sein

strlen(V4814) > 50

C7314 [W] Medikament 14: Angabe darf max. 50 Zeichen lang sein

strlen(V4815) > 50

C7901 [E] Das SGI-Reservefeld 1 ist noch nicht definiert. Es muss leer sein.

strlen(V4405) > 0

C7902 [E] Das SGI-Reservefeld 2 ist noch nicht definiert. Es muss leer sein.

strlen(V4406) > 0

C7910 [E] Das 16. Reservefeld ist noch nicht definiert. Es muss leer sein.

strlen(V4816) > 0

C7911 [E] Reservefeld 16: Angabe darf max. 50 Zeichen lang sein

strlen(V4816) > 50

C7920 [E] Das 17. Reservefeld ist noch nicht definiert. Es muss leer sein.

strlen(V4817) > 0

C7921 [E] Reservefeld 17: Angabe darf max. 50 Zeichen lang sein

strlen(V4817) > 50

C7930 [E] Das 18. Reservefeld ist noch nicht definiert. Es muss leer sein.

strlen(V4818) > 0

C7931 [E] Reservefeld 18: Angabe darf max. 50 Zeichen lang sein

strlen(V4818) > 50

C7940 [E] Das 19. Reservefeld ist noch nicht definiert. Es muss leer sein.

strlen(V4819) > 0

C7941 [E] Reservefeld 19: Angabe darf max. 50 Zeichen lang sein

strlen(V4819) > 50

C7950 [E] Das 20. Reservefeld ist noch nicht definiert. Es muss leer sein.

strlen(V4820) > 0

C7951 [E] Reservefeld 20: Angabe darf max. 50 Zeichen lang sein

strlen(V4820) > 50

D. Indikatoren

D0. Meldungen

D0000 [C] Anzahl Fälle total

$1 == 1$

D0001 [C] Datensatz enthält Fehler

c_ECount > 0

D0002 [C] Datensatz enthält Warnungen

c_WCount > 0

D0003 [C] Datensatz enthält Hinweise

c_HCount > 0

D0004 [C] Datensatz enthält Fehler oder Warnungen

c_ECount +c_WCount > 0

D0005 [C] Datensatz enthält Fehler, Warnungen oder Hinweise

c_ECount + c_WCount + c_HCount > 0

D0101 [C] Datensatz enthält Fehler

c_ECount > 0

D0102 [C] Datensatz enthält Warnungen

c_WCount > 0

D0103 [C] Datensatz enthält Hinweise

c_HCount > 0

D0104 [C] Datensatz enthält Fehler oder Warnungen

c_ECount +c_WCount > 0

D0105 [C] Datensatz enthält Fehler, Warnungen oder Hinweise

c_ECount + c_WCount + c_HCount > 0

D0201 [C] Kantonale Zusatzdaten vorhanden

hasMK

ACTION: hasMK=FALSE; V0304="0";

D1. Patienten

D1000 [C] Anzahl Fälle (Statistiktyp A): Austritte während der Beobachtungsperiode

strcmp(V0202,'A') == 0

D1001 [C] Anzahl Fälle (Statistiktyp B): Eintritte während des Jahres und am 31.12 hospitalisiert

strcmp(V0202,'B') == 0 && strcmp(V1301,'3') == 0

D1002 [C] Anzahl Fälle (Statistiktyp C): Langzeitaufenthalte

strcmp(V0202,'C') == 0 && strcmp(V1301,'3') == 0

D1010 [C] Anzahl Austritte (stationäre A-Fälle)

strcmp(V0202,'A') == 0 && strcmp(V1301,'3') == 0

D1011 [C] Ambulanter Fall

strcmp(V1301,'1') == 0

ACTION: doWriteRec=FALSE;

D1012 [C] Zusammensetzung nach Behandlungsart: Anzahl teilstationäre Patienten

strcmp(V0202,'A') == 0 && strcmp(V1301,'2') == 0

D1020 [C] Zusammensetzung nach Geschlecht: Anzahl weibliche Patienten

strcmp(V0202,'A') == 0 && strcmp(V1101,'2') == 0

D1030 [C] Zusammensetzung nach Alter (Statistiktyp A): Mittelwert Alter

strcmp(V0202,'A') == 0

D2. Eintritt

D2008 [C] Aufenthalt vor Eintritt 8 andere (unpräzise Angabe)

strcmp(V0202,'A') == 0 && strcmp(V1202,'8') == 0

D2009 [C] Aufenthalt vor Eintritt 9 unbekannt (unpräzise Angabe)

strcmp(V0202,'A') == 0 && strcmp(V1202,'9') == 0

D2018 [C] Eintrittsart 8 andere (unpräzise Angabe)

strcmp(V0202,'A') == 0 && strcmp(V1203,'8') == 0

D2019 [C] Eintrittsart 9 unbekannt (unpräzise Angabe)

strcmp(V0202,'A') == 0 && strcmp(V1203,'9') == 0

D2028 [C] Einweisende Instanz 8 andere (unpräzise Angabe)

strcmp(V0202,'A') == 0 && strcmp(V1204,'8') == 0

D2029 [C] Einweisende Instanz 9 unbekannt (unpräzise Angabe)

strcmp(V0202,'A') == 0 && strcmp(V1204,'9') == 0

D3. Aufenthalt

D3009 [C] Liegeklasse 9 unbekannt (unpräzise Angabe)

strcmp(V0202,'A') == 0 && strcmp(V1302,'9') == 0

D3018 [C] Hauptkostenträger 8 andere (unpräzise Angabe)

strcmp(V0202,'A') == 0 && strcmp(V1402,'8') == 0

D3019 [C] Hauptkostenträger 9 unbekannt (unpräzise Angabe)

strcmp(V0202,'A') == 0 && strcmp(V1402,'9') == 0

D3029 [C] Behandlungsart 9 unbekannt (unpräzise Angabe)

strcmp(V0202,'A') == 0 && strcmp(V1301,'9') == 0

D3100 [C] Zusammensetzung nach Aufenthaltsdauer: Mittlere stat. Aufenthaltsdauer

strcmp(V0202,'A') == 0 && strcmp(V1301,'3') == 0

D4. Austritt

D4005 [C] Entscheid für Austritt: Todesfall

strcmp(V0202,'A') == 0 && strcmp(V1502,'5') == 0

D4008 [C] Entscheid für Austritt: 8 andere (unpräzise Angabe)

strcmp(V0202,'A') == 0 && strcmp(V1502,'8') == 0

D4009 [C] Entscheid für Austritt: 9 unbekannt (unpräzise Angabe)

strcmp(V0202,'A') == 0 && strcmp(V1502,'9') == 0

D4018 [C] Aufenthalt nach Austritt: 8 andere (unpräzise Angabe)

strcmp(V0202,'A') == 0 && strcmp(V1503,'8') == 0

D4019 [C] Aufenthalt nach Austritt: 9 unbekannt (unpräzise Angabe)

strcmp(V0202,'A') == 0 && strcmp(V1503,'9') == 0

D4028 [C] Behandlung nach Austritt: 8 andere (unpräzise Angabe)

strcmp(V0202,'A') == 0 && strcmp(V1504,'8') == 0

D4029 [C] Behandlung nach Austritt: 9 unbekannt (unpräzise Angabe)

strcmp(V0202,'A') == 0 && strcmp(V1504,'9') == 0

D5. Diagnosen

D5000 [C] Hauptdiagnose: Angabe vorhanden

strcmp(MB->V0202,'A') == 0 && !DIA[0].isNull

D5001 [C] Diagnose: vorhanden

strcmp(MB->V0202,'A') == 0 && !DIA[i].isNull

D5002 [C] Zusatz zur Hauptdiagnose: Angabe vorhanden

strcmp(MB->V0202,'A') == 0 && !DIA[1].isNull

D5003 [C] Nebendiagnose: Angabe vorhanden

i > 1 && strcmp(MB->V0202,'A') == 0 && !DIA[i].isNull

D6. Behandlungen

D6000 [C] Hauptbehandlung: Angabe vorhanden

strcmp(MB->V0202,'A') == 0 && !BEH[0].isNull

D6001 [C] Behandlung: vorhanden

strcmp(MB->V0202,'A') == 0 && !BEH[i].isNull

Anhang A. ICD/CHOP Metadaten-Fileformat

Die Prüfung der Diagnose- und Operationskodes erfolgt mit Hilfe von Metadaten. Es handelt sich dabei um eine Tabelle für die ICD-10 und eine Tabelle für die CHOP.

In den Plausibilitätstests wird mit den Funktionen INDEXCOL(Kode,Col) jeweils auf die entsprechende Kolonne zugegriffen. Die genaue Beschreibung der Funktionen befindet sich in Anhang C.

ICD-10:

1. Kode	2. KodeTyp	3. Komplementärkode	4. Sex	5. AlterMin	6. AlterMax
Liste der 3-/4-stelligen ICD-10 GM-Kodes, die für die Tests verwendet werden	0 = Äussers Ursache / ohne Kap. Z 1 = Absoluter + Kode mit * (1:1) im Titel 2 = Absoluter * Kode (1:n) 3 = Anderer, konventioneller Kode 4 = Absoluter + ohne * im Titel 5 = Zusatz aus Kap. XX erforderlich 6 = Unspezifischer 3-stelliger ICD-Kode	Gültige Komplementärkodes zum entsprechenden Stern-/Kreuzkode	Zulässiges Geschlecht für Kodevergabe. Leer, falls nicht geschlechtsspezifisch	Zulässige untere Altersgrenze für Vergabe der Kodes	Zulässige obere Altersgrenze für Vergabe der Kodes

Die Kodes des Typs 6 können grösstenteils durch gültige vierstellige Kodes mit Endung .9 ersetzt werden. In diesen Fällen erscheint im Feld Komplementärkode der entsprechende vierstellige Kode.

CHOP:

1. Kode	2. KodeTyp	3. Komplementärkode	4. Sex	5. AlterMin
Liste der 2-, 3-, und 4-stelligen CHOP-Kodes.	1 = nicht definiert 2 = unspezifischer 2-Steller 3 = unspezifischer 3-Steller 4 = gültiger 4-Steller 5 = gültiger 3-Steller 6 = gültiger 6-Steller 7 = 5-Steller	Liste der gültigen Komplementärkodes. Zur Zeit keine Einträge.	Zulässiges Geschlecht für Kodevergabe. Leer, falls nicht geschlechtsspezifisch 1 = nur Männer 2 = nur Frauen	Noch nicht definiert.

6. AlterMax	7. ICD-10 Hauptdiagnose	8. gültig ab	9. gültig bis	10. Ersatzcode
Noch nicht definiert.	Angabe der Hauptdiagnose, die dieser Behandlung entspricht. Noch nicht definiert.	Jahrzahl, ab der der Kode verwendet werden darf.	Jahrzahl, bis und mit der der Kode verwendet werden darf.	Kode, in den ein ungültiger/revidierter Kode überführt wird.

Anhang B. Beschreibung der in den Tests verwendeten Funktionen

Die in den vorhergehenden Kapiteln beschriebenen Tests verwenden eine Anzahl von Funktionsaufrufen, die im folgenden beschrieben werden. Generell orientieren sich die MedPlaus-Tests an der C++-Syntax

=
Zuweisung var = Wert (is)
V0301='0';

==
entspricht, bei Vergleich (equal)
strlen(V0002) == 0

!=
entspricht nicht, bei Vergleich (not equal)
strlen(V0008) != 0

<
kleiner als (less than)
atoI(V0008) < 3

>
grösser als (greater than)
strlen(V0103) > 2

!
nicht (NOT)
!(dateV0004.isNull)

&&
und (AND)
strlen(V0009) != 0 && !isHex(V0009)

||
oder (OR)
!(dateV0004.isNull || dateV0004.isYYYYMMDD)

+
Addition
(MB->STAYH+15)

*****
Multiplikation
*(dateV1501.DateValue – dateV1201.DateValue +1) * 24*

/
Division
(atol(V2302)/10) > 44

for ()
For-Schleife. Führt Anweisung aus, solange Bedingung in For-Schleife erfüllt ist
++ zählt hoch

for (int di=0; di <=DIACount;di++)

->
Verweist auf Record, der Variable beinhaltet („Pointer“)
strcmp(MB->V0202,'A') == 0

atoi
Wandelt String- in Integerwerte um (StringToInt)
isInt(V1103) && atoi(V1103)

atoI
Wandelt String- in Integerwerte um (StringToInt)
atoI(V0008) < 3

BEH[pos]
Behandlungsfunktionen. Prüfen die Behandlungsvariablen
pos gibt die Position an (0 = HB, 1 = 1. NB, 2 = 2. NB, 3 = 3. NB etc.)
.isNull: Behandlungsfeld leer

.Data[CHOP_]: Attribute des Codes
.Key: Feldinhalt
BEH[0].isNull

datevar
Datumsfunktionen. Prüfen die Datumsvariable
.isNull: Datumsfeld leer
.isYYYYMMDD: Datumsfeld im Format JJJJMMTT
.isYYYY0000: Datumsfeld im Format JJJJ0000
.DateValue: Datumswert
.hasDate: enthält gültiges Datum
.hasYear: enthält gültiges Jahr
.hasMonth: enthält gültigen Monat 1 – 12
.hasHour: enthält gültige Stunde 0 – 23
.hasMinute: enthält gültige Minute 0 - 59
.Hour: Stundenwert
!(dateV0004.isNull)

DIA[pos]
Diagnosefunktionen. Prüfen die Diagnosevariablen
pos gibt die Position an (0 = HD, 1 = ZHD, 2 = 1. ND, 3 = 2. ND etc.)
.isNull: Diagnosefeld leer

.Data[ICD_]: Attribute des Codes
.Key: Feldinhalt
strlen(DIA[0].Key) > 0

DIACount
Zählt die Anzal Diagnosen eines Records
for (int di=0; di <=DIACount;di++)

doWriteRec
Steuert, ob der Record geschrieben wird
= FALSE (nein)
= TRUE (ja)
doWriteRec=FALSE;

hasDiaLike('DIA')
Prüft, ob Record die angegebene Diagnose enthält, die mit 'DIA'beginnt
!hasDiaLike('Z38')

hasMN
Prüft, ob Record einen Neugeborenen-Zusatzdatensatz enthält
!hasMN

hasMP
Prüft, ob Record einen Psychiatrie-Zusatzdatensatz enthält
hasMP=FALSE

hasF64
Prüft, ob Record F64-Diagnose enthält
!hasF64

hasZ38_147
Prüft, ob Record Z38-Diagnose enthält
!hasZ38_147

BOOL hasKeyLikeLen(char* key, int len, char* keylist)
Prüft, ob substr(key,0,len) in der Liste enthalten ist. Die Listenelemente sind mit Pipe abgetrennt.
hasKeyLikeLen(DIA[i].Key,3,'|Z42|Z44|Z47|Z48|Z50|Z51|Z54|')

isAlphaNumUpperCase(var)
Prüft, ob der String nur Zeichen der Menge ['0'..'9','A'..'Z'] enthält
!isAlphaNumUpperCase(V4601)

isExklDia(DIA[pos].Key)
Prüft das Vorhandensein einer Reihe von spezifischen, festgelegten Diagnosecodes
isExklDia(DIA[0].Key)

isHex(var)
Liefert den booleschen Wert true, falls die Angabe var hexadezimal ist
!isHex(V0009)

isIn(var,nr,_list)
Liefert den logischen Wert des Resultats der Suche des Wertes var in einer indizierten Liste (true: gefunden, false: nicht gefunden)
!isIn(V0202,3_ABC)

isInt(var)
Liefert den booleschen Wert true, falls die Angabe var eine Ganzzahl ist
isInt(V1103)

isMKCH
Prüft, ob Record ein kantonaler Zusatzdatensatz MK ist
isMKCH && strlen(V5102) == 0

isSSDia(DIA[pos].Key)
Prüft ob die Diagnose eine Schwangerschaftsdiagnose ist: O03-O07; O10-O16; O20-O29; O30-O48; O60-O75; O80-O82; Z34-Z37
isSSDia(DIA[i].Key)

BOOL isSubKeyIn(char *str, int offset, int len, int size, char* array[])
SubKey = substr(str,offset,len). Liefert Wahr, wenn das Array mit size Einträgen einen Wert enthält, der mit SubKey beginnt.
isSubKeyIn(DIA[i].Key,1,2,16,_16_DIA_PERINAT_LEN2)

LineNumberInMX
Zählt die Anzahl der gelieferten Datensätze
LineNumberInMX > atol(MX->V0008)

STAYH
Aufenthaltsdauer in Stunden
(MB->STAYH+15)

char * strchr(char * str, int character)
Lokalisiert das erste Auftreten eines bestimmten Zeichens in einer Zeichenkette
strchr('0123456',V2302[2]) == NULL

strcmp(var,'value')
Liefert das Resultat eines Vergleichs von 2 Zeichenketten (StringCompare)
strcmp(V1104,'CHE') == 0

strncmp(var,'value',pos)
Liefert das Resultat eines Vergleichs einer bestimmten Position von 2 Zeichenketten (StringCompare)
strncmp(DIA[0].Key,'Z38',3) != 0

strlen(var)
Liefert die Anzahl Zeichen, die in var enthalten sind (StringLength)
strlen(V0002) == 0

var_list.inIndex
Prüft, ob der Variablenwert in einer vorgegebenen Indexliste enthalten ist
!(V1104_regio.inIndex)

Bemerkung
ACTION
Funktionen hinter der Bezeichnung ACTION: werden nur im speziellen Korrekturmodus von MedPlaus ausgeführt. Und dies auch nur, wenn die spezifische Testbedingung erfüllt ist.
V0301='0';