

Beschreibung der Befehlsstruktur des in der Simulationsstudie verwendeten Programms  
(Rasch-Test zu 3 Messgelegenheiten)

**basic**

Dieser Befehl startet das Systat BASIC Programm.

**new**

Mit diesem Befehl wird ein neuer Datensatz erzeugt.

**repeat 10000**

Dieser Befehl reserviert 10000 Zeilen für die Zufallszahlen.

**let ksi=zrn**

Dieser Befehl definiert ksi als standardnormalverteilte  $[N(0,1)]$  latente Trait-Variable.

**let sig=sqr(0.3\*xrn(1))**

Mit diesem Befehl wird sig (die personspezifische Standardabweichung über Messgelegenheiten) als Quadratwurzel einer mit 0.3 multiplizierten  $\chi^2$ -verteilten Zufallsgröße (mit einem Freiheitsgrad) definiert.

**dim tau(3)**

Dieser Befehl reserviert drei Variablen für die drei Messgelegenheiten.

**for i=1 to 3**

**let tau(i)=ksi+zrn(0,sig)**

Mit tau (1..3) werden die drei latenten State-Variablen bezeichnet. Jede einzelne dieser drei Variablen setzt sich aus dem ksi-Wert und dem Wert einer normalverteilten Zufallsgröße mit dem Mittelwert null und der personspezifischen Standardabweichung sig zusammen.

**next**

**dim it(129)**

Dieser Befehl reserviert drei mal 43 Variablen für die drei Antwortmuster.

**for j=1 to 3**

**for i=1 to 43**

**let it((j-1)\*43+i) = 1/(1+exp(((i-22)/7)-tau(j))) > urn**

**next**

Mit den ersten beiden Befehlszeilen werden Schleifen über alle drei Messgelegenheiten und alle 43 Items definiert.

Der Ausdruck  $(j-1)*43+i$  zeigt auf ein Item (1..129).

Der Ausdruck  $(i-22)/7$  ergibt die Itemschwierigkeit. (Für das erste Item  $i=1$  ergibt sich eine Schwierigkeit von  $-3$ , für das letzte Item  $i=129$  eine Schwierigkeit von  $+3$ , so dass die Schwierigkeiten insgesamt im Bereich von  $-3$  bis  $+3$  liegen.)

Der Ausdruck  $1/(1+\exp((i-22)/7)-\tau(j))$  ist die logistische Wahrscheinlichkeit für eine richtige Antwort gegeben die Schwierigkeit  $(i-22)/7$  und die Fähigkeit  $\tau(j)$ .

Der Ausdruck urn definiert eine gleichverteilte (0,1) Zufallsgröße.

Der Gesamtausdruck  $1/(1+\exp((i-22)/7)-\tau(j)) > \text{urn}$  ergibt somit 0, wenn falsch und 1, wenn richtig geantwortet wurde.

**next**

**run**