

**1 Erstellen Sie jeweils ein Struktogramm mit der Kontrollstruktur Sequenz**

- 1.1. Der Benutzer soll 2 Zahlen eingeben. Damit sollen nacheinander die 4 Grundrechenarten ausgeführt werden. Das Ergebnis soll jeweils ausgegeben werden.
- 1.2. Aus 3 Einzelnoten ist ein Notendurchschnitt zu berechnen. Die Noten werden von Tastatur erfasst, der Schnitt am Bildschirm ausgegeben
- 1.3. Ein Kunde bestellt 3 verschiedene Artikel. Die Menge und der Nettoeinzelpreis werden jeweils von Tastatur erfasst. Am Ende wird der Mehrwertsteueranteil und die Gesamtsumme inkl. Mehrwertsteuer ausgegeben (Der Mehrwertsteuersatz soll für alle Artikel 19% betragen).
- 1.4. Ein EURO-Betrag (Benutzereingabe) soll in Dollar umgerechnet werden. Der Umtauschkurs wird im Programm fest programmiert (siehe aktueller Kurs im Internet). EURO-Betrag, DOLLAR-Betrag und Umtauschkurs sind am Bildschirm auszugeben.
- 1.5. Sie bestellen sich zusammen mit n Freunden eine Partypizza (n als Benutzereingabe). Die Pizza ist in 16 Stücke unterteilt. Jeder von Ihnen soll gleichviele Stücke bekommen. Damit sich keiner ungerecht behandelt fühlt, sollen die restlichen Stücke an ihren Hund verfüttert werden. Schreiben Sie ein Programm das ausgibt, wie viele Stücke ihr Hund bekommt!
- 1.6. Erstellen Sie ein Programm, das die Anzahl der Scheine für die Ausgabe eines Geldautomaten berechnet. Der Benutzer soll dabei einen Geldbetrag eingeben, den er auszahlt bekommen möchte.  
Im Geldautomat sind folgende Banknoten verfügbar:  
50€, 20€, 10€ und 5€ Scheine.  
Der Geldautomat zahlt den Betrag so aus, dass möglichst wenige Scheine benötigt werden.  
**Beispiel:** Wenn Sie 130€ abheben möchten, gibt ihnen der Geldautomat folgende Banknoten aus: 2 x 50€, 1 x 20€ und 1 x 10€.
- 1.7. Die Gesamtlänge der Tour de France 2009 betrug 3460 km mit insgesamt 21 Etappen. Erstellen Sie ein Struktogramm, das folgende Berechnungen durchführt und die Ergebnisse am Bildschirm ausgibt.
  - a. Wie lang war eine Etappe im Schnitt?
  - b. Wie viel Prozent der Gesamtstrecke entspricht eine Durchschnittsetappe?
  - c. Der Sieger benötigte insgesamt 85:48:35 Std. Der Drittplazierte Lance Armstrong benötigte 5:24 Min. länger.  
Wie hoch war die Durchschnittsgeschwindigkeit von Lance ?
  - d. Es wurden 20 Teams eingeladen, jedes Team besteht aus 9 Fahrern. 24 Fahrer schieden vorzeitig aus. Wieviel Prozent der Fahrer erreichten das Ziel in Paris?
- 1.8. Aus der Körpergröße in Metern und dem Körpergewicht (Masse) einer Person in kg soll der Body-Mass-Index berechnet werden. Er wird folgendermaßen berechnet:

$$\text{BMI} = \frac{\text{Körpergewicht}}{(\text{Körpergröße})^2}$$

Die Größen werden von der Tastatur eingelesen, der BMI am Bildschirm ausgegeben.

**2 Erstellen Sie ein Struktogramm mit der Kontrollstruktur Verzweigung:**

- 2.1. Der Lagerbestand eines Artikels soll geprüft werden. Ist er unter dem Mindestlagerbestand, soll die Bestellmenge ermittelt und am Bildschirm ausgegeben werden: es müssen so viele Artikel bestellt werden, dass die doppelte Mindestbestandsmenge am Lager ist. Lagerbestand und Mindestlagerbestand werden von Tastatur eingegeben.
- 2.2. Rabattvergabeprüfung:  
Bei einem Bestellwert ab 500 EURO werden 10 % Rabatt gewährt, zwischen 200 EURO und 499,99 5 %, sonst kein Rabatt. Rabattbetrag und Rechnungsbetrag sollen ausgegeben werden.
- 2.3. Preisvergabe: Aus 3 Einzelnoten ist ein Notendurchschnitt zu berechnen. Liegt dieser besser oder gleich 1,4 gibt es einen Preis, bei besser oder gleich 1,8 eine Belobigung. Besser als 4,0 gilt als bestanden, der Rest hat nicht bestanden.
- 2.4. Blutgruppenverträglichkeitsprüfung:

		Spender			
		A	B	AB	0
Empfänger	A	v	n	n	v
	B	n	v	n	v
	AB	v	v	v	v
	0	n	n	n	v

Erstellen Sie ein Struktogramm, das nach Eingabe der Spender- und Empfängerblutgruppe entweder "verträglich" oder "unverträglich" ausgibt. Wenden Sie logische Operatoren an, um die Zahl der Verzweigungen möglichst gering zu halten.

- 2.5. Erstellen Sie ein Struktogramm, das eine Ganzzahl-Note nach vorgegebener Notenskala einstuft und das Ergebnis am Bildschirm ausgibt. Der Benutzer soll über Tastatur zunächst eine Ganzzahl-Note zwischen 1 und 6 eingeben. Das Programm gibt den entsprechenden Notentext aus:

a: Ganzzahl-Note	b: Gleitpunktzahl-Note	Notenskala
1	$1,0 \leq \text{Note} < 1,5$	sehr gut
2	$1,5 \leq \text{Note} < 2,5$	gut
3	$2,5 \leq \text{Note} < 3,5$	befriedigend
4	$3,5 \leq \text{Note} < 4,5$	ausreichend
5	$4,5 \leq \text{Note} < 5,5$	mangelhaft
6	$5,5 \leq \text{Note} < 6,0$	ungenügend
Sonst	Sonst	Falsche Eingabe!

- 2.6. Lösen Sie dasselbe Problem für eine Gleitpunktzahl-Note.
- 2.7. Zu bestimmen ist die größte von 3 unterschiedlichen Zahlen, die von Tastatur eingegeben werden.

- 2.8. Es soll ein Struktogramm für ein Rechenprogramm erstellt werden, das die 4 Grundrechenarten ausführt.  
Einzugeben sind 1.Operand, Operator, 2. Operand. Gültige Operatoren sind +, -, \* oder /. Division durch 0 ist zu verhindern, ungültige Operatoren sind abzufangen. Das Rechenergebnis wird am Bildschirm ausgegeben.
- 2.9. Das Struktogramm zum Body-Mass-Index aus Aufgabe 1.8 soll erweitert werden.  
In Abhängigkeit vom errechneten Wert soll eine Auswertung erfolgen:

Kategorie	BMI
Starkes Untergewicht	< 16
Untergewicht	< 18,5
Normalgewicht	18,5 – 25
Übergewicht	> 25
starkes Übergewicht	> 40

### 3 Erstellen Sie ein Struktogramm mit der Kontrollstruktur **Aufruf**

Allgemein: Überlegen Sie, ob es sich bei der jeweiligen Aufgabenstellung um den Entwurf eines *aufrufenden* oder eines *aufgerufenen* Struktogramms handelt. Handelt es sich um ein aufgerufenes Struktogramm, so erstellen Sie dieses und zusätzlich ein geeignetes aufrufendes Struktogramm mit einem Ablauf Ihrer Wahl.

- 3.1. Das „aufrufende“ Struktogramm „berechneSchnitt“ liest drei Noten von der Tastatur ein. Die Mittelwertberechnung soll in einem extra Struktogramm erfolgen („aufgerufenes“ Struktogramm). Die Einzelnoten werden als Parameter übergeben. Der Mittelwert kommt als Rückgabewert zurück. Erstellen Sie das aufgerufene Struktogramm mit Titel `berechneNotenSchnitt`.

Titel:	berechneSchnitt
Parameter:	keine
Rückgabe:	keine
Kurzbeschreibung: erfasst 3 Noten von Tastatur und errechnet den Schnitt	
A: "Bitte geben Sie die erste Note ein:"	
E: note1	
A: "Bitte geben Sie die zweite Note ein:"	
E: note2	
A: "Bitte geben Sie die dritte Note ein:"	
E: note3	
schnitt ← berechneNotenSchnitt(note1, note2, note3)	
A: "Notenschnitt =" schnitt	

- 3.2. Von der Tastatur wird ein Text erfasst. Wenn dieser Text mehr als 50 Zeichen enthält geben Sie den Text „Das ist ein langer Text mit xxx Zeichen“ aus, ansonsten den Text „Das ist ein kurzer Text mit xxx Zeichen“ aus.  
Anmerkung: zur Ermittlung der Länge eines Textes steht das Struktogramm mit Titel `ermittleLänge` zur Verfügung (ist gegeben, muss nicht selbst erstellt werden), das den zu prüfenden Text als Parameter erhält und die Anzahl der Zeichen als Rückgabewert liefert.
- 3.3. **Passwortüberprüfung:**  
ein eingegebenes Passwort soll mindestens 8 Zeichen lang sein. Ist es kürzer, soll die Meldung "Passwort zu kurz" kommen, ansonsten "Passwort noch einmal eingeben". Falls dieses Passwort gleich ist wie das zuerst eingegebene kommt "Passwort ok", sonst "Passwort stimmt nicht überein".  
Anmerkung: Wiederverwendung von `ermittleLänge`.
- 3.4. Ändern Sie das Struktogramm „Preisvergabe“ (Nr. 2.3, Verzweigung) so ab, dass die Ermittlung der Durchschnittsnote in einem eigenen Struktogramm mit Titel `berechneNotenSchnitt` (siehe Aufg. 3.1) erfolgt, das die Durchschnittsnote als Rückgabewert liefert.

3.5. Bonitätsprüfung:

Ein Unternehmen prüft bei jedem Auftragseingang die Bonität des Kunden und macht davon die Zahlungsbedingungen abhängig. Bei A-Kunden wird ein Zahlungsziel von 3 Wochen und 2 % Skonto eingeräumt, bei B-Kunden 2 % Skonto, bei C-Kunden zahlbar sofort. Erstellen Sie einen Text für die entsprechenden Zahlungsbedingungen und geben Sie den Text aus.

Das Struktogramm soll Teil eines größeren Softwarepakets sein. Es erhält als Übergabeparameter eine Kundennummer und ruft seinerseits ein Struktogramm „ermittleBonität“ auf. Dieses hat den Parameter: Kundennummer und die Rückgabe: Bonität ('A', 'B' oder 'C').

3.6. Für Kalenderfunktionen (z.B. in Tabellenkalkulationsprogrammen oder Datenbanken) ist es wichtig zu wissen, ob ein Jahr ein Schaltjahr ist, oder nicht.

Für Schaltjahre gelten folgende Regeln:

- Jahre, die nicht durch 4 ohne Rest teilbar sind, sind keine Schaltjahre
- Ein Jahr, das durch 4, durch 100 und durch 400 ohne Rest teilbar ist, ist ein Schaltjahr.
- Ein Jahr, das durch 4 und 100 (und nicht durch 400) ohne Rest teilbar ist, ist kein Schaltjahr.
- Ein Jahr, das durch 4 (und nicht durch 100) ohne Rest teilbar ist, ist ein Schaltjahr.

Das Struktogramm des Moduls „istSchaltjahr()“ erhält als Parameter eine Jahreszahl und gibt als Rückgabewert „wahr“ zurück, falls es sich um eine Schaltjahr handelt, sonst „falsch“

#### 4 Erstellen Sie jeweils ein Struktogramm mit der Kontrollstruktur **Wiederholung**

Allgemein: Setzen Sie für die folgenden Aufgaben die aus Ihrer Sicht jeweils am besten geeignete Schleifenform ein: die kopfgesteuerte, die fußgesteuerte- oder die Zählschleife

##### 4.1 Zahleneingabe von Tastatur:

- alle eingegebenen Zahlen sollen addiert werden, bis der Benutzer eine Zahl  $< 0$  eingibt. Daraufhin wird die Gesamtsumme ausgegeben.
- es sollen so lange Zahlen von Tastatur eingegeben werden, bis die Gesamtsumme von 1000 überschritten wird. Anschließend ist die Anzahl der Summanden auszugeben.

##### 4.2 Erweitern Sie das Struktogramm "berechneSchnitt", das in Aufgabe 2.3 der Verzweigungsübung verwendet wird. Es können beliebig viele Noten erfasst werden. Die Noteneingabe ist beendet, wenn der Benutzer "-1" eingibt.

##### 4.3 Passwortüberprüfung:

ein Passwort soll so lange angefordert werden, bis der Benutzer die Mindestlänge von 8 Zeichen eingegeben hat oder die Länge 0 für Ende.  
Ist es kürzer als 8 Zeichen, soll die Meldung "Passwort muss mindestens 8 Zeichen lang sein" kommen, ansonsten "Passwort noch einmal eingeben". Falls dieses Passwort gleich ist wie das zuerst eingegebene kommt "Passwort ok", sonst "Passwort stimmt nicht überein".

##### 4.4 Auf Grund des hohen Jahresgewinns soll an die Mitarbeiter eine Prämie verteilt werden. Sie hängt ab von Firmenzugehörigkeit, Anzahl Fehltage und Kinderzahl.

Die Prämie wird folgendermaßen vergeben:

Ist ein Mitarbeiter zum Stichtag 1.11.2018 3 Jahre im Betrieb und hat er weniger als 5 Fehltage, so erhält er eine Prämie in Höhe von 10 % seines Gehalts, bei 5 und mehr Fehltagen 5 %.

Ist er weniger als 3 Jahre im Betrieb und hat er weniger als 5 Fehltage, so erhält er 5 % Prämie. Bei 5 und mehr Fehltagen erhält er eine Prämie von 3 %, aber nur wenn er mehr als 2 Kinder hat.

Die Personalnummer und die ermittelte Prämie sind am Bildschirm auszugeben.

Grundlage für die Berechnung ist die Tabelle MITARBEITER, die folgenden Aufbau hat:

Personalnummer (pnr)

Kinderzahl (kzahl)

AnzahlFehltage (ftage)

Eintrittsdatum (edat, Format: jjjjmmmtt)

Gehalt (gehalt)

Übernehmen Sie den Inhalt des jeweils nächsten Datensatzes aus der Tabelle durch die Anweisung `datensatz ← liesDatensatz()`. Die genannten Datenfelder sind darauf hin verfügbar. Ist das Ende der Tabelle erreicht, liefert die Methode `liesDatensatz()` als Rückgabewert die symbolische Konstante EOF (End of File), statt einem Datensatz.

- 4.4 Ein Programm soll ermitteln, wie viele Schaltjahre es im Zeitraum von 1900 bis 2018 gab. Es wird davon ausgegangen, dass es den Modul (die Methode) `istSchaltjahr()` (siehe Aufg. 3.6) bereits gibt.
- 4.6 Erstellen Sie ein Struktogramm für ein Modul (Methode) zur Berechnung der Fakultät einer ganzen Zahl (Parameter: eine natürliche Zahl inkl. 0; Rückgabe: Fakultät der Zahl).

Zur Erläuterung: die „Fakultät“ ist für positive, ganze Zahlen definiert und errechnet sich aus der Multiplikation der ganzen Zahlen von 1 bis  $n$ . Die Schreibweise ist  $n!$ .

$n\_fakultaet = n! = 1 * 2 * 3 * \dots * (n-1) * n$

( $n \geq 0$ , ganzzahlig), wobei gilt:  $0! = 1$

Beispiel:

$5! = 1 * 2 * 3 * 4 * 5$

- 4.7 Zahlenraten 1 aus 100  
Mit einem Zufallszahlengenerator ist zunächst eine zu erratende ganze Zahl aus dem Bereich  $1 \leq \text{Zahl} \leq 100$  zu bestimmen. (Anweisung:  $\text{Zahl} \leftarrow \text{zZahl}(1, 100)$ ). Nach jedem Rateversuch wird ein Hinweis ausgegeben, ob die gesuchte Zahl größer oder kleiner ist. Sobald die Zahl erraten ist wird die Anzahl der Versuche ausgegeben und bewertet: 1 – 2: "Super", 3 – 4: "Sehr gut"; 5 – 6: "Gut"; 7 – 8: "Mittelmäßig"; > 8: "schwach".
- 4.8 ggT-Berechnung nach Euklid:  
Der Modul (Methode) „`berechneGGT`“, bekommt als Parameter zwei natürliche Zahlen, deren größter gemeinsamer Teiler als Rückgabewert zurück gegeben werden soll.

Der (moderne) euklidische Algorithmus führt dazu in jedem Schritt eine Division mit Rest aus. Der Rest wird im nächsten Schritt zum neuen Divisor. Der Divisor, bei dem der Rest 0 wird ist der ggT.

Beispiel:

38 : 10 = 3 Rest 8

10 : 8 = 1 Rest 2

8 : 2 = 4 Rest 0

ggT von 38 und 10 ist 2.



## 4.9 Test des Zufallszahlengenerators:

Nachdem ein Würfel 500 Mal geworfen wurde ( $\text{Zahl} \leftarrow \text{ZZahl}(1, 6)$ ), soll die absolute und relative Häufigkeit für das Eintreten der Zahl 6 ausgegeben werden. Entwickeln Sie eine Struktogramm, das diesen Sachverhalt abbildet.

## 4.10 Zahlenraten

Der Benutzer denkt sich eine Zahl zwischen 1 und 100. Der Computer gibt eine Zahl aus. Ist die Zahl zu groß, gibt der Benutzer 'g' für „zu groß“ ein. Ist die Zahl zu klein, gibt der Benutzer 'k' für „zu klein“ ein. Danach macht der Computer einen neuen Rateversuch und bekommt vom Benutzer wieder einen entsprechenden Tipp. Hat der Computer die Zahl getroffen ist die Eingabe 'r' für „richtig“. Bei einem optimalen Algorithmus errät der Computer die Zahl in maximal 7 Schritten.

## 4.11 Ein Algorithmus soll ein Zahlendreieck der Form

```
1
22
333
4444
55555
```

ausgeben. Die Anzahl der Zeilen soll als Parameter gegeben sein (Bereich 1...9)

Hinweis: Das Dreieck durch Ausgabe einzelner Zeichen erzeugen. Ein Zeilenvorschub wird mit A: <Zeilenvorschub> erreicht.

## 4.12 Ein Algorithmus soll ein Kästchen der Form

```
+---+
++--
+++
++++
```

ausgeben. Die Anzahl der Zeilen soll als Parameter gegeben sein (Bereich 1...50)

Hinweis: Das Kästchen durch Ausgabe einzelner Zeichen erzeugen. Die Ausgabe eines Pluszeichens wird mit A: '+' erzeugt. Ein Zeilenvorschub wird mit A: <Zeilenvorschub> erreicht.

## 4.13 Kredittilgung

Sie sind Programmierer in einer Bank. Zur Unterstützung der Kundenberater sollen Sie ein Programm entwickeln, das einen Tilgungsplan für einen Kredit errechnet und ausgibt.

In das Programm werden Kreditbetrag, Zinssatz und die monatliche Tilgungsrate eingegeben. Die Höhe der Tilgungsrate bleibt über die gesamte Laufzeit gleich (sie setzt sich zusammen aus Zins + Tilgung).

In einer Liste sollen dann die Werte Monat, Restdarlehen, Zins und Tilgung und die Zinssumme ausgegeben werden, damit ersichtlich ist, nach wie vielen Monaten das Darlehen abbezahlt ist, wie sich Zins und Tilgung entwickeln und wieviel die Gesamtzinsen betragen.



## Zusätzliche Übungsaufgaben

4.14 Schreiben Sie ein Programm, das eine Zinseszinstabelle ausgibt. Nach Eingabe des Einzahlungsbetrags soll eine Tabelle dargestellt werden, aus der man ersieht, wie viel Euro man nach 1 bis 5 Jahren bei Zinssätzen von 2, 3, 4, 5 und 6 Prozent „angespart“ hat. (Zum Lösen dieser Aufgabe benötigen Sie keine Zinseszinsformel – verzinsen Sie einfach das Kapital des jeweiligen Vorjahres. In den Spalten werden die Jahre von 1 bis 5 aufgeführt und in den Zeilen untereinander die Zinssätze von 2 bis 6 Prozent!!!)

4.15 Stellen Sie sich vor, Sie haben im Lotto eine Million Euro gewonnen!

Von diesem Gewinn wollen Sie leben, ohne zu arbeiten!

Dazu wollen Sie ein Programm erstellen, das Ihnen die Anzahl der Jahre ausgibt, die Sie von dem Gewinn leben können, ohne zu arbeiten, indem Sie sich von diesem Gewinn jährlich ein „Gehalt“ auszahlen.

- Sie legen den Gewinn auf einem Sparbuch an und erhalten jährlich Zinsen. Den Zinssatz soll der Benutzer eingeben!
- Nun soll der Benutzer eingeben, welches „Gehalt“ er jährlich vom Gewinn ausgezahlt bekommen möchte.
- Im Programm soll für jedes Jahr der übriggebliebene Betrag des Gewinns nach der Verzinsung und dem Abzug des Gehalts berechnet werden.
- Am Ende soll die Anzahl der Jahre ausgegeben werden, für die das Geld reicht!  
**Achtung:** Um eine Endlosschleife zu vermeiden, soll die Schleife auf jeden Fall abbrechen, wenn 100 Jahre erreicht sind!

4.16 Entwickeln Sie ein Struktogramm, das mit nur vier Variablen ein Dreieck beliebiger Größe auf dem Bildschirm ausgibt. Falls Sie Probleme mit dieser Aufgabe haben sollten:

- a. Füllen Sie diese Tabelle für die dargestellte Bildschirmausgabe aus.
- b. Finden Sie den Zusammenhang zwischen der Zeilennummer, der Anzahl der Leerzeichen bzw. Sternchen und ggf. der Höhe heraus.
- c. Bei Bedarf werten Sie eine Tabelle für ein Dreieck der Höhe 5 aus etc.

Bildschirmausgabe		
Bitte Hoehe eingeben: 4		
<pre>       *      ***     *****    ***** </pre>		
Zeilen- nummer	Anzahl der Leerzeichen	Anzahl der Sternchen
1		
2		
3		
4		

4.17 Ändern Sie das Programm aus Aufgabe 4.16 so, dass das Dreieck auf dem Kopf stehend auf dem Bildschirm dargestellt wird. (Wenn Sie Aufgabe 4.16 mit nur vier (4!) Variablen gelöst haben, müssen Sie nur eine Programmzeile ändern;)

Bildschirmausgabe		
Bitte Hoehe eingeben: 4		
<pre> *****  *****   *****    *****     * </pre>		