


Arbeitsblatt Nr. 7	TAF 11.3: Strukturierte Programmierung	 B S G G
Datum:	Thema: Entscheidungen	
Seite 1 von 4	Name:	

Entscheidungen

Verzweigungen im Programmablauf werden durch Entscheidungen ermöglicht. Eine Entscheidung enthält eine Bedingung (die man auch als eine Frage formulieren kann). Diese Bedingung ist entweder erfüllt (wahr/true) oder nicht erfüllt (falsch/false). Man könnte auch sagen, dass die Antwort auf die „Frage“ „ja“ oder „nein“ lautet.

Bedingungen werden mit Hilfe der Vergleichsoperatoren (siehe folgende Seite) formuliert.

Bedingte Anweisung

Der einfachste Fall ist die bedingte Anweisung. Hierbei wird eine Bedingung formuliert und wenn diese Bedingung erfüllt ist, werden zusätzliche Befehle im Programmablauf ausgeführt.

```
if ( Bedingung )
{
  Befehlsfolge...
}
```

```
if ( alter < 18 )
{
  Console.WriteLine("Nicht volljährig");
}
```

Hinter der Bedingung (erste Zeile) folgt ein einzelne Anweisung oder ein Block von Anweisungen. Bei einer einzelnen Anweisung können die geschweiften Klammern entfallen. Geschweifte Klammern dienen dazu, einen Block von zusammen gehörigen Anweisungen zu definieren, einen sogenannten „Anweisungsblock“.

Entscheidung

Bei der Entscheidung werden hinter der Bedingung alternative Anweisungen angegeben. Ist die Bedingung erfüllt, werden die Anweisungen im Anweisungsblock hinter der Bedingung ausgeführt, ansonsten werden die Anweisungen hinter **else** ausgeführt.

```
if ( Bedingung )
{
  ...Befehle
}
else
{
  andere Befehle
}
```

```
if ( alter < 18 )
{
  Console.WriteLine("Nicht volljährig");
}
else
{
  Console.WriteLine("Volljährig");
}
```

Mehrere Entscheidungen

Man kann auch mehrere Entscheidungen, die dann nacheinander überprüft werden, formulieren. Sobald eine der Entscheidungen zutrifft, somit also „wahr“ ist, wird der zugehörige Anweisungsblock ausgeführt und im Anschluss wird das Programm hinter der Entscheidungsstruktur fortgesetzt.

```
if ( Bedingung_1 )
{
  ...Befehle
}
else if ( Bedingung_2 )
{
  andere Befehle
}
else
{
  nochmal andere Befehle
}
```

```
if ( alter < 14 )
{
  Console.WriteLine("Kind");
}
else if ( alter >= 14 && alter < 18)
{
  Console.WriteLine("Jugendlicher");
}
else
{
  Console.WriteLine("Volljähriger");
}
```

Mehrfachentscheidung (Auswahl)

Mittels Auswahl kann eine Variable auf verschiedene bekannte Werte überprüft werden. Die auszuwertende Variable wird hinter dem Befehlswort **switch** in runden Klammern angegeben. In einem nun folgenden Anweisungsblock werden die zu überprüfenden Werte für diese Variable hinter dem Befehlswort **case** angegeben. Für jeden zu überprüfenden Fall kann nun eine Folge von Befehlen angegeben werden, die ausgeführt wird, wenn der Fall zutrifft.

Jede dieser Befehlsfolgen wird mit dem Befehlswort **break** beendet. Fehlt das Befehlswort **break**, werden die Befehle in nachfolgenden Befehlsfolgen ebenfalls ausgeführt. Durch das Befehlswort **break** wird die Auswahlstruktur verlassen und die Befehlsausführung hinter dem **switch**-Anweisungsblock fortgesetzt.

Am Ende der Auswahlstruktur befindet sich häufig der **default**-Fall. Dieser wird ausgeführt, wenn keiner der vorher definierten Fälle zutrifft. Der **default**-Fall muss nicht definiert werden!

```

switch ( Variable )
{
  case wert1:
    Befehlsfolge 1
    break;

  case wert2:
    Befehlsfolge 2
    break;

  ...
  ...
  default:
    Befehle...
}

```

```

int auswahl;
auswahl = Convert.ToInt32(Console.ReadLine());
...
switch ( auswahl )
{
  case 1:
    Befehlsfolge 1
    break;

  case 2:
    Befehlsfolge 2
    break;

  ...
  ...
  default:
    Befehle...
}

```


Die auszuwertende Variable muss ein Ganzzahltyp (**int**, **char**, **bool**) sein. Es ist **nicht möglich**, Wertebereiche der Variable per Auswahl zu überprüfen.

Die Auswahl wird verwendet, wenn eine Variable nur ganz bestimmte, im Vorhinein bekannte Werte, annehmen kann. Ein klassisches Beispiel hierfür ist die Erstellung eines Menüs mit mehreren Menüpunkten zur Auswahl durch den Benutzer. Der **default**-Fall wäre dann gegeben, wenn der Benutzer einen nicht vorhandenen Menüpunkt auswählt.

Vergleichsoperatoren

Zur Formulierung von Bedingungen werden sogenannte Vergleichsoperatoren verwendet. Diese sind:

Identität	Ungleichheit	Größer	Kleiner	Größer gleich	Kleiner gleich
==	!=	>	<	>=	<=

Arbeitsblatt Nr. 7	TAF 11.3: Strukturierte Programmierung		B S G G
Datum:	Thema: Entscheidungen		
Seite 3 von 4	Name:		

Bedingte Logikoperatoren

Eine Bedingung kann aus mehreren Teil-Bedingungen bestehen, die mittels bedingter Logikoperatoren verknüpft werden. Diese Logikoperatoren sind:

- das logische Und (&&)
- das logische Oder (||)
- das logische Nicht (!).

Beispiele zur Anwendung der Logikoperatoren in Bedingungen

- `if (alter >= 14 && alter < 18)`
Beide Teilbedingungen (`alter >= 14`) und (`alter < 18`) müssen zutreffen, damit die Bedingung „wahr“ ist (die Variable `alter` hat im Beispiel den Typ `int`).

- `if (buchstabe == 'a' || buchstabe == 'A')`
Wenn die Variable `buchstabe` (vom Typ `char`) das kleine a oder das große A als Wert enthält, ist die Bedingung „wahr“. In diesem Beispiel schließen sich beide Werte aus, `buchstabe` kann nur einen der beiden Werte annehmen.

```
if ( alter >= 21 || buchstabe == 'A' )
```

Hier ist die Bedingung „wahr“, wenn eine der beiden Teil-Bedingungen oder sogar beide Teil-Bedingungen „wahr“ sind.

- `if (!volljährig)`
Hat eine Variable den Datentyp `bool`, kann das logische Nicht angewendet werden. (die Variable `volljährig` soll hier den Typ `bool` haben)


```
if ( !(alter >= 18) )
```

Bei Ausdrücken, auf die der Nicht-Operator angewendet werden soll, muss der betreffende Ausdruck geklammert werden.

Bei der Verknüpfung von Teil-Bedingungen mit Logik-Operatoren empfiehlt es sich mit Klammern zu arbeiten. Ähnlich wie bei arithmetischen Operatoren, gibt es auch bei den Logik-Operatoren eine Rangfolge hinsichtlich der Auswertung eines solchen Ausdrucks¹.

Den höchsten Rang hat hierbei das logische Nicht, den nächsthöchsten Rang hat das bedingte logische Und. Den niedrigsten Rang hat das bedingte logische Oder:

¹ Siehe <https://docs.microsoft.com/de-de/dotnet/csharp/language-reference/operators/boolean-logical-operators> abgerufen am 25.02.2020
© Uwe Homm Version vom 25. Februar 2020

Arbeitsblatt Nr. 7	TAF 11.3: Strukturierte Programmierung	 B S G G
Datum:	Thema: Entscheidungen	
Seite 4 von 4	Name:	

Übungen

Erstellen Sie für die nachfolgenden Übungen eine leere Projektmappe „Entscheidungen“ und fügen Sie für jede Aufgabe ein Projekt hinzu.

Skizzieren Sie für jede Aufgabe ein Struktogramm!

1. Erstellen Sie ein Programm, das eine Division von zwei Gleitkommazahlen durchführt und das Ergebnis ausgibt. Sollte der Divisor jedoch Null sein, ist die Berechnung nicht durchzuführen und stattdessen soll eine Fehlermeldung „*Division durch Null nicht möglich*“ angezeigt werden.
2. Mitarbeiter sollen eine Gehaltserhöhung bekommen. Die Gehaltserhöhung soll 2,5%, mindestens jedoch 150 € betragen.
Erstellen Sie ein Programm, das den aktuellen Wert des Gehalts einliest und den neuen Wert entsprechend berechnet.
3. Es ist ein neues Steuersystem eingeführt worden. Alle Einkünfte ab 100.000 € pro Jahr sollen mit dem Steuersatz 46% besteuert werden. Alle darunter liegenden Einkünfte sollen mit 23% besteuert werden.
Erstellen Sie ein Programm, welches das Einkommen einliest und die jeweils zu zahlenden Steuern ausgibt.
4. Da die Besteuerung in Aufgabe 4 für kleine Einkommen ungerecht ist, sollen erst ab einem Einkommen von 10.000 € pro Jahr Steuern anfallen.
Verwenden Sie das Programm aus Aufgabe 4 und modifizieren Sie es entsprechend.
5. Zwei Geraden in einer Ebene mit den Gleichungen $y = a * x + b$ und $y = c * x + d$ sind durch die Werte von a, b, c und d bestimmt. Schreiben Sie ein Programm, das die Lage der beiden Geraden zueinander analysiert und ihren Schnittpunkt ausgibt, falls er existiert!
6. Ein Programm soll die Umrechnungen Euro → Dollar, Dollar → Euro, Euro → Pfund und Pfund → Euro durchführen. Mit Hilfe eines Menüs soll die gewünschte Umrechnung ausgewählt werden. Wenn ein nicht vorhandener Menüpunkt ausgewählt wird, soll eine Hinweismeldung erscheinen.