

# Das Modul Text

- Modul
- Length und Empty
- GetChar
- FromChar und Sub
- SetChars und FromChars
- Zusammenfassung

## Das Modul Text - 1

- Für die Bearbeitung vieler Problemstellungen ist es notwendig, Operationen auf Text durchzuführen.
- In Modula-3 wird der Datentyp TEXT zur Verfügung gestellt. Objekte dieses Typs sind endliche Zeichenfolgen (Strings genannt). Sie müssen in Anführungsstrichen (" ") geschrieben werden.
- Ein Großteil der Operationen auf dem Typ TEXT liefert das Bibliotheks-Modul Text.

## Das Modul Text - 2

- Die wichtigsten Operationen ermöglichen:
  - Bestimmung der Länge eines Strings
  - Auswahl von Zeichen aus einem String
  - Auswahl von Teil-Strings
  - Umwandlung von Zeichen in Strings
  - Umwandlung zwischen Zeichen-Array und String

## Das Text Interface

Ausschnitt aus dem Interface mit den wichtigsten Operationen:

```
INTERFACE Text;
TYPE T =TEXT; (* Text.T entspricht TEXT *)
PROCEDURE Cat(t, u: T): T;
PROCEDURE Equal(t, u: T): BOOLEAN;
PROCEDURE GetChar(t: T; i: CARDINAL): CHAR;
PROCEDURE Length(t: T): CARDINAL;
PROCEDURE Empty(t: T): BOOLEAN;
PROCEDURE Sub(t: T; start, length: CARDINAL): T;
PROCEDURE SetChars(VAR a: ARRAY OF CHAR; t: T);
PROCEDURE FromChar(c: CHAR): T;
PROCEDURE FromChars(READONLY a: ARRAY OF CHAR): T;
...
END Text.
```

## Length und Empty

```
VAR t: TEXT;
    i: CARDINAL;
...
t := SIO.GetLine();
...
IF Text.Length(t) = i THEN ... END;
...
IF Text.Empty(t) THEN ... END;
...
```

### ■ Length

- liefert zu einem String die Anzahl der enthaltenen Zeichen, d.h. die Länge des Strings
- die gelieferte Zahl ist vom Typ CARDINAL

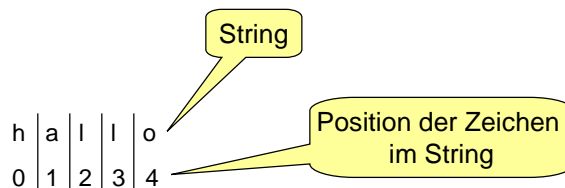
### ■ Empty

- liefert TRUE für einen gegebenen String str, wenn  $\text{Text.Length}(\text{str}) = 0$

## Position von Zeichen

- Die Position eines Zeichens in einem String wird durch eine Zahl vom Typ **CARDINAL** repräsentiert.
- Das erste Zeichen hat die Position 0, das zweite die Position 1, ... .
- Wenn  $\text{Text.Length}(\text{str})$  gleich  $n$  ist ( $n > 0$ ), dann hat der das letzte Zeichen die Position  $(n-1)$ .

### ■ Beispiel:



## Zeichenauswahl

```
VAR t: TEXT;
    c: CHAR;
    i: CARDINAL;
...
t := SIO.GetLine();
...
c := Text.GetChar(t, i);
...
```

### ■ GetChar

- erhält einen String und eine Zahl vom Typ CARDINAL
- ein Fehler tritt auf, wenn die gegebene Zahl größer oder gleich der Länge des Strings ist oder kleiner als
- liefert das Zeichen, welches im String an der Position der Zahl steht
- Beispiel: Text.GetChar("hallo", 2)

## FromChar

```
VAR s: TEXT;
    c: CHAR;
...
c := SIO.GetChar();
...
s := Text.FromChar(c);
...
```

### ■ FromChar

- erhält ein Zeichen (Typ: CHAR) als Eingabe
- liefert einen String (Typ: TEXT) zurück, der nur aus dem angegebenen Zeichen besteht

## Sub - 1

```
VAR s, t: TEXT;
    x, y: CARDINAL;
...
s := SIO.GetLine();
...
t := Text.Sub(s, x, y);
...
```

### ■ Sub

- erhält einen String und zwei Zahlen (erste Zahl: x, zweite Zahl: y) vom Typ CARDINAL als Eingabe
- liefert den String zurück, der aus den y Zeichen besteht, die bei Position x beginnen
- Beispiel: Text.Sub("hallo", 1, 2) liefert den String „al“.

## Sub - 2

### ■ Die Operation Sub hat mehrere zu beachtende Spezialfälle:

- Wenn eine der beiden Zahlen kleiner als 0 ist, tritt ein Fehler auf.
- Der leere String wird zurückgegeben, wenn x (die Position) größer gleich der Länge des Strings ist oder y = 0 (die Länge des auszugebenden Strings).
- Wenn der eingegebene String nicht genügend Zeichen enthält (d.h.,  $x + y > \text{Text.Length}(\text{string})$ ), dann wird der bei x beginnende (und bis zum Ende des Strings reichende) Teil-String ausgegeben.

## SetChars und FromChars

- Diese beiden Prozeduren dienen der **Konvertierung** zwischen Strings und Arrays mit Elementen vom Typ CHAR.
- **SetChar**
  - erhält ein Array mit Elementen vom Typ Char (im Folgenden array genannt) und einen String (im folgenden string genannt)
  - füllt array so lange mit Elementen von string, bis array voll ist oder kein Zeichen mehr in string übrig
  - Füllt string array nicht komplett auf, so bleiben die übrigen Elemente unverändert
  - array muß veränderbar sein (d.h. nicht read-only)
- **FromChar**
  - erhält ein Array mit Elementen vom Typ CHAR
  - liefert einen String, der aus den Elementen des Arrays besteht

## Zusammenfassung - 1

- **Modul Text**
  - dient der Bearbeitung von Text
  - hier wurde nur der wichtigste Teil vorgestellt
  - das komplette Interface befindet sich in der Datei Text.i3
- **Verschiedene Operationen**
  - ermöglichen die Betrachtung der Länge eines Strings
  - ermöglichen den Zugriff auf einzelne Zeichen oder Zeichenfolgen im String
  - ermöglichen die Konvertierung in / aus "verwandten" Typen

## Zusammenfassung - 2

---

### ■ weitere Operationen

- liefern zu einer Hash-Funktion zu einem String
- vergleichen zwei Strings bzgl. der lexikographischen Ordnung
- liefern zu einem String und einem Zeichen die Position des ersten Auftretens des Zeichens in dem String
- liefern zu einem String und einem Zeichen die Position des letzten Auftretens des Zeichens in dem String