

Informatik I

Tobias Joschko, Reiner Hüchting

DHBW Mannheim

WS 2020

Datentypen

Datentypen

Maschinen-Datentypen

- ▶ bit: einzelne 0 oder 1
- ▶ byte: kleinste adressierbare Einheit, üblicherweise 8 Bit
- ▶ word: von Maschinencode verarbeitete Einheit, 32 oder 64 Bit

Datentypen

Maschinen-Datentypen

- ▶ bit: einzelne 0 oder 1
- ▶ byte: kleinste adressierbare Einheit, üblicherweise 8 Bit
- ▶ word: von Maschinencode verarbeitete Einheit, 32 oder 64 Bit

Zahlen

- ▶ int: ganze Zahlen, Speicherung als word, manchmal byte
- ▶ float: Gleitkommazahlen, Speicherung in word

Datentypen

Maschinen-Datentypen

- ▶ bit: einzelne 0 oder 1
- ▶ byte: kleinste adressierbare Einheit, üblicherweise 8 Bit
- ▶ word: von Maschinencode verarbeitete Einheit, 32 oder 64 Bit

Zahlen

- ▶ int: ganze Zahlen, Speicherung als word, manchmal byte
- ▶ float: Gleitkommazahlen, Speicherung in word

Sonstige

- ▶ char: Repäsentiert Zeichen (Buchstaben), meistens byte
- ▶ bool: Wahr oder Falsch, 0/1 oder Zahl mit Falsch = 0.

Datentypen

Maschinen-Datentypen

- ▶ bit: einzelne 0 oder 1
- ▶ byte: kleinste adressierbare Einheit, üblicherweise 8 Bit
- ▶ word: von Maschinencode verarbeitete Einheit, 32 oder 64 Bit

Zahlen

- ▶ int: ganze Zahlen, Speicherung als word, manchmal byte
- ▶ float: Gleitkommazahlen, Speicherung in word

Sonstige

- ▶ char: Repäsentiert Zeichen (Buchstaben), meistens byte
- ▶ bool: Wahr oder Falsch, 0/1 oder Zahl mit Falsch = 0.

Diese Typen heißen **primitiv**, die folgenden **zusammengesetzt**.

Datentypen

Array

- ▶ Zusammenhängender Block im Speicher, der mehrere Elemente vom gleichen Typ enthält.

Datentypen

Array

- ▶ Zusammenhängender Block im Speicher, der mehrere Elemente vom gleichen Typ enthält.
- ▶ Beispiele:

Datentypen

Array

- ▶ Zusammenhängender Block im Speicher, der mehrere Elemente vom gleichen Typ enthält.
- ▶ Beispiele:
 - ▶ Zeichenketten (`string`): Array aus `char`

Datentypen

Array

- ▶ Zusammenhängender Block im Speicher, der mehrere Elemente vom gleichen Typ enthält.
- ▶ Beispiele:
 - ▶ Zeichenketten (`string`): Array aus `char`
 - ▶ Text: Array aus Zeilen (jede Zeile ein `string`)

Datentypen

Array

- ▶ Zusammenhängender Block im Speicher, der mehrere Elemente vom gleichen Typ enthält.
- ▶ Beispiele:
 - ▶ Zeichenketten (`string`): Array aus `char`
 - ▶ Text: Array aus Zeilen (jede Zeile ein `string`)
 - ▶ Notation für Zugriff meist mit eckigen Klammern:
`text[i]` liefert den i -ten Eintrag des Arrays `text`.

Datentypen

Array

- ▶ Zusammenhängender Block im Speicher, der mehrere Elemente vom gleichen Typ enthält.
- ▶ Beispiele:
 - ▶ Zeichenketten (`string`): Array aus `char`
 - ▶ Text: Array aus Zeilen (jede Zeile ein `string`)
 - ▶ Notation für Zugriff meist mit eckigen Klammern:
`text[i]` liefert den i -ten Eintrag des Arrays `text`.

Listen

- ▶ Allgemeinere Form des Arrays.
- ▶ Muss nicht zusammenhängend im Speicher stehen.
- ▶ Notation in Pseudocode wie Arrays.

Datentypen

Verkettete Listen

Datentypen

Verkettete Listen

- ▶ einfach verkettete Listen

Datentypen

Verkettete Listen

- ▶ einfach verkettete Listen
- ▶ doppelt verkettete Listen

Datentypen

Verkettete Listen

- ▶ einfach verkettete Listen
- ▶ doppelt verkettete Listen

Graphen und Bäume

Datentypen

Verkettete Listen

- ▶ einfach verkettete Listen
- ▶ doppelt verkettete Listen

Graphen und Bäume

- ▶ weitere Verallgemeinerung von Listen, Elemente können mehrere Nachfolger/Vorgänger haben

Datentypen

Verkettete Listen

- ▶ einfach verkettete Listen
- ▶ doppelt verkettete Listen

Graphen und Bäume

- ▶ weitere Verallgemeinerung von Listen, Elemente können mehrere Nachfolger/Vorgänger haben
- ▶ Baum: Jedes Element hat 0 oder 1 Vorgänger

Datentypen

Verkettete Listen

- ▶ einfach verkettete Listen
- ▶ doppelt verkettete Listen

Graphen und Bäume

- ▶ weitere Verallgemeinerung von Listen, Elemente können mehrere Nachfolger/Vorgänger haben
- ▶ Baum: Jedes Element hat 0 oder 1 Vorgänger
- ▶ Graph: beliebig komplexe Verbindungsstruktur

Datentypen

Verkettete Listen

- ▶ einfach verkettete Listen
- ▶ doppelt verkettete Listen

Graphen und Bäume

- ▶ weitere Verallgemeinerung von Listen, Elemente können mehrere Nachfolger/Vorgänger haben
- ▶ Baum: Jedes Element hat 0 oder 1 Vorgänger
- ▶ Graph: beliebig komplexe Verbindungsstruktur

Zentrales Konzept: Pointer

- ▶ Variablen, die Speicheradressen enthalten.

Datentypen

Assoziative Arrays

Datentypen

Assoziative Arrays

- ▶ Speichern Paare aus *Schlüsseln* und *Werten* (engl. **key-value-pairs**).

Datentypen

Assoziative Arrays

- ▶ Speichern Paare aus *Schlüsseln* und *Werten* (engl. **key-value-pairs**).
- ▶ Alternative zum Array, Zugriff nicht über einen Index, sondern über den Schlüssel.

Datentypen

Assoziative Arrays

- ▶ Speichern Paare aus *Schlüsseln* und *Werten* (engl. **key-value-pairs**).
- ▶ Alternative zum Array, Zugriff nicht über einen Index, sondern über den Schlüssel.

Themen in diesem Kurs

Datentypen

Assoziative Arrays

- ▶ Speichern Paare aus *Schlüsseln* und *Werten* (engl. **key-value-pairs**).
- ▶ Alternative zum Array, Zugriff nicht über einen Index, sondern über den Schlüssel.

Themen in diesem Kurs

- ▶ Arrays, Strings und Pointer

Datentypen

Assoziative Arrays

- ▶ Speichern Paare aus *Schlüsseln* und *Werten* (engl. **key-value-pairs**).
- ▶ Alternative zum Array, Zugriff nicht über einen Index, sondern über den Schlüssel.

Themen in diesem Kurs

- ▶ Arrays, Strings und Pointer
- ▶ Entwurf von einfachen eigenen Datentypen

Datentypen

Assoziative Arrays

- ▶ Speichern Paare aus *Schlüsseln* und *Werten* (engl. **key-value-pairs**).
- ▶ Alternative zum Array, Zugriff nicht über einen Index, sondern über den Schlüssel.

Themen in diesem Kurs

- ▶ Arrays, Strings und Pointer
- ▶ Entwurf von einfachen eigenen Datentypen
- ▶ Bäume und Graphen im nächsten Semester.

Datentypen

Binärdarstellung von ganzen Zahlen

- ▶ Bits sind Ziffern zur Basis 2

Datentypen

Binärdarstellung von ganzen Zahlen

- ▶ Bits sind Ziffern zur Basis 2
- ▶ Zahlen sind Listen von Bits

1	1	0	0	1
---	---	---	---	---

Datentypen

Binärdarstellung von ganzen Zahlen

- ▶ Bits sind Ziffern zur Basis 2
- ▶ Zahlen sind Listen von Bits

1	1	0	0	1
---	---	---	---	---

- ▶ zählen Vorkommen der 2er-Potenzen
- ▶ von rechts anfangen bei 2^0

Datentypen

Binärdarstellung von ganzen Zahlen

- ▶ Bits sind Ziffern zur Basis 2
- ▶ Zahlen sind Listen von Bits

1	1	0	0	1
---	---	---	---	---

- ▶ zählen Vorkommen der 2er-Potenzen
- ▶ von rechts anfangen bei 2^0

$$1 \times 2^0$$

Datentypen

Binärdarstellung von ganzen Zahlen

- ▶ Bits sind Ziffern zur Basis 2
- ▶ Zahlen sind Listen von Bits

1	1	0	0	1
---	---	---	---	---

- ▶ zählen Vorkommen der 2er-Potenzen
- ▶ von rechts anfangen bei 2^0

$$1 \times 2^0 + 0 \times 2^1$$

Datentypen

Binärdarstellung von ganzen Zahlen

- ▶ Bits sind Ziffern zur Basis 2
- ▶ Zahlen sind Listen von Bits

1	1	0	0	1
---	---	---	---	---

- ▶ zählen Vorkommen der 2er-Potenzen
- ▶ von rechts anfangen bei 2^0

$$1 \times 2^0 + 0 \times 2^1 + 0 \times 2^2$$

Datentypen

Binärdarstellung von ganzen Zahlen

- ▶ Bits sind Ziffern zur Basis 2
- ▶ Zahlen sind Listen von Bits

1	1	0	0	1
---	---	---	---	---

- ▶ zählen Vorkommen der 2er-Potenzen
- ▶ von rechts anfangen bei 2^0

$$1 \times 2^0 + 0 \times 2^1 + 0 \times 2^2 + 1 \times 2^3$$

Datentypen

Binärdarstellung von ganzen Zahlen

- ▶ Bits sind Ziffern zur Basis 2
- ▶ Zahlen sind Listen von Bits

1	1	0	0	1
---	---	---	---	---

- ▶ zählen Vorkommen der 2er-Potenzen
- ▶ von rechts anfangen bei 2^0

$$1 \times 2^0 + 0 \times 2^1 + 0 \times 2^2 + 1 \times 2^3 + 1 \times 2^4$$

Datentypen

Binärdarstellung von ganzen Zahlen

- ▶ Bits sind Ziffern zur Basis 2
- ▶ Zahlen sind Listen von Bits

1	1	0	0	1
---	---	---	---	---

- ▶ zählen Vorkommen der 2er-Potenzen
- ▶ von rechts anfangen bei 2^0

$$\begin{array}{ccccccccc} 1 \times 2^0 & + & 0 \times 2^1 & + & 0 \times 2^2 & + & 1 \times 2^3 & + & 1 \times 2^4 \\ = & 1 & & + & & & 8 & + & 16 \end{array}$$

Datentypen

Binärdarstellung von ganzen Zahlen

- ▶ Bits sind Ziffern zur Basis 2
- ▶ Zahlen sind Listen von Bits

1	1	0	0	1
---	---	---	---	---

- ▶ zählen Vorkommen der 2er-Potenzen
- ▶ von rechts angefangen bei 2^0

$$\begin{aligned} & 1 \times 2^0 + 0 \times 2^1 + 0 \times 2^2 + 1 \times 2^3 + 1 \times 2^4 \\ = & 1 + 8 + 16 \\ = & 25 \end{aligned}$$

Datentypen

Aufgabe: Geben Sie Algorithmen bzw. Programme an, die eine Zahl von der Binärdarstellung ins Dezimalsystem konvertieren und umgekehrt.

Datentypen

Binär nach Dezimal

EINGABE: Binärzahl x

AUSGABE: Die Variable ergebnis

```
ergebnis = 0
```

```
for i from length(x) to 1:  
    ergebnis = ergebnis + x[i] * 2i
```

Datentypen

Dezimal nach Binär

EINGABE: Dezimalzahl x

AUSGABE: Die Variable ergebnis

```
ergebnis = [0]
```

```
i = 1
```

```
while x != 0:
```

```
    ergebnis[i] = x mod 2
```

```
    x = x div 2
```

```
    i = i + 1
```

```
ergebnis umdrehen
```