

#### 4.2 Worin besteht das Prinzip von Johann von Neumann? Welche Komponenten besitzt ein Rechner, der nach diesem Prinzip arbeitet?

grundsätzlichere Bedeutung, da verschiedenartige Steuerungsvorgänge abstrahiert und auf einheitliche Verarbeitungsmethoden zurückgeführt werden.

Prinzip: Nicht nur Speicherung von Nutzinformationen sondern auch das Programm wird als Information gespeichert,  
→ Konzept des speicherprogrammierbaren Rechners

Aufbau:

Input, output

Bus

Prozessoren

Speicher

#### 4.4 Erklären Sie die Merkmale der objektorientierten Sprache C++!

1. Abstrakte Datentypen,
  2. Vererbung,
  3. Polymorphismus (Vielgestaltigkeit).
- + Erklärung!

#### Was bedeutet ein Datenmodell? Wie kann man es in C/C++ abbilden?

Variablen und Konstanten bilden das Datenmodell als Träger der Informationen.

Variablen- und Typenvereinbarungen als Einheit bilden das rechnerinterne Datenmodell

Datenmodelle = Zusammenfassung aller Datenobjekte → DM = Zusammenfassung aller Informationen in einer Programmeinheit

→ Datentypen = benennbare Speichereinheiten eines bestimmten Datentyps → Beispiele

#### Welche Erweiterungen kennt C++ im Vergleich zu C?

Erweiterung von C um objektorientierte Konzepte

→ ein weiterer Typ = Klasse → beinhaltet Attribute und Methoden

→ Realisierung als Kapselung, Vererbung und Polymorphismus

#### Wie erfolgt die Übergabe und Rückgabe mehrerer Parameter über Funktionen?

Call by reference...

#### Wie und mit welcher Vorgehensweise kann man in C Daten dauerhaft speichern?

Abruf oder Hineinschreiben von Daten auf einem permanenten Speicher aus einem Programm heraus immer über Pufferbereich im Arbeitsspeicher

Puffer nimmt eine größere Menge von Daten auf, damit nicht jedes Datenobjekt separat zwischengespeichert werden muss.

→ Für dieses Wechselspiel ist strukturierter Typ FILE festgelegt:

```
typedef struct {
    char *buffer; /* Zeiger für die Adresse des Dateipuffers */
    char *ptr; /* Zeiger auf das nächste Zeichen im Puffer */
    int cnt; /* Anzahl der Zeichen im Puffer */
    int flags; /* Bits mit Angaben zum Dateistatus */
    int fd; /* Deskriptor (Kennzahl der Datei) */
} FILE;
```

Erklären Sie an einem Beispiel die einfache Vererbung in C++!

Klasse nur Komponenten von einer anderen Klasse erbt, spricht man von einfacher Vererbung daß eine Klasse die Komponenten einer anderen Klasse, also Eigenschaften und Methoden, erbt und gleichzeitig neue Komponenten hinzugefügt werden können

## Beispiel 9.8

Definition einer Klasse *Base* und Ableitung einer weiteren Klasse *Inherit*.

```
class Base { // Oberklasse
private:
    int k;
    char *ptr;
public:
    int method();
};

class Inherit : public Base {
double f;
};
```

### Welche Konzepte verwendet die objektorientierte Programmierung?

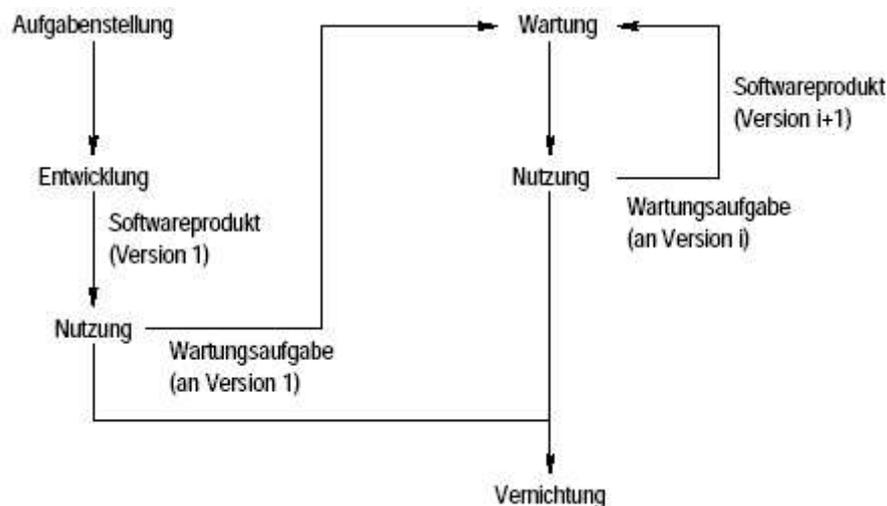
Die objektorientierte Programmierung erweitert die abstrakten Datentypen um das Konzept der Typ-/Untertyp-Beziehungen zwischen einzelnen Klassen.  
→ durch Vererbung

Abstrakte Datentypen

Polymorphismus

Vererbung

### Welche Phasen durchläuft ein Softwareprodukt?



### 4.3 Was sind Zeiger? Nennen Sie Anwendungsfelder!

= Zugriff auf den Speicherplatz einer Variablen über Adressen  
enthalten Anfangsadressen von Variablen, d.h.

Verwendung:

- \_ direkt mit Adressen rechnen,
- \_ man kann Funktionen mit mehreren Ergebnisparametern organisieren
- \_ Felder = Zeiger auf das erste Feldelement.
- \_ Aufbau dynamische Datenstrukturen