

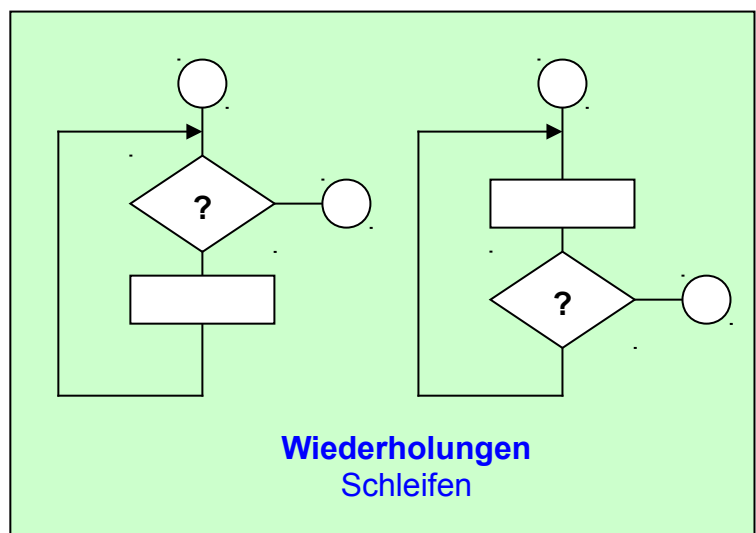
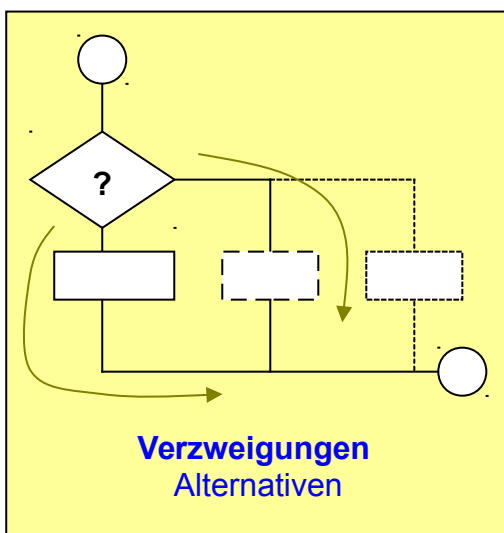
Kontrollstrukturen / Ablaufsteuerung

Nur bei sehr einfachen Übungsprogrammen ist der Programmablauf geradlinig und sequentiell. Ein reales Programm benutzt **Kontrollstrukturen**, um den konkreten Ablauf aufgrund von Nutzereingaben oder Zustandsabfragen steuern. Wie in C und C++ lassen sich zwei grundlegende Arten der Ablaufsteuerung unterscheiden:

- **Verzweigung** in jeweils unterschiedliche Programmabschnitte (Alternative)
- **Wiederholung** eines bestimmten Teilabschnitts (Schleife)

In beiden Gruppen gibt es mehrere Unterarten, die durch unterschiedliche **Schlüsselworte** definiert werden.

Wenn der Programmablauf auf eine solche Kontrollanweisung trifft, dann wird der **Bedingungs-Ausdruck** abgearbeitet. Hier können Programmzustände (meist Inhalte von Variablen) abgefragt. Die zugehörigen Aktionen werden in einem eigenen Anweisungsblock { } definiert.



Die im folgenden Kapitel gewählte Darstellung beruht auf dem standardisierten Organisationsmodell des **Programmablaufplans** (PAP), der besonders für die visuelle Repräsentation von vollständigen Programmen geeignet ist, während Struktogramme eher für detaillierte Darstellung einzelner Kontrollstrukturen eingesetzt werden.

Programmablaufpläne sind allgemein gut geeignet, um komplexe Ablaufstrukturen abzubilden. Sie werden häufig bei prozeduralen Sprachen wie C verwendet. Objektorientierte Programmierung verlangt darüber hinaus den Einsatz von Klassenmodellen, die nicht den eigentlichen Programmablauf modellieren, sondern vorwiegend das Verhältnis von Klassen bzw. Objekten zueinander darstellen. Hierfür ist die **UML** das geeignete Modellierungswerkzeug.