

9 Programmierung

9.1 Schritte zur algorithmischen Lösung von Problemen

- **Problemdefinition**

Beispiel: Volumenberechnung einer Kugel
Eingabe: Radius **rad**
Ausgabe: Volumen **vol**

- **Entwurf eines Algorithmus zur Problemlösung (Struktogramm)**

$vol = \frac{4}{3} \pi rad^3$

- **Formulierung des Algorithmus in einer Programmiersprache**

```
//Kugel.cpp
//Berechnung des Volumens einer Kugel
#include <iostream.h>

void main()
{
    const double PI = 3.141592;           //Beginn von main
    double rad, vol;                     //Konstantenvereinbarung
    cout << "Bitte Radius eingeben";     //Variablenvereinbarung
    cout << endl;                          //Ausgabe von Text
    cout << "Radius = ";                   //neue Zeile
    cin >> rad;                             //Eingabe des Radius
    vol = 4.0/3 * PI *rad * rad *rad;      //Wertzuweisung
    cout << "Das Volumen betraegt " << vol; //Ausgabe des Volumens
    cout << endl;
}                                           //Ende von main
```

Anmerkungen:

- Allgemeiner Aufbau eine C++-Quelldatei (ASCII-Datei, die beim Speichern automatisch die Endung *.cpp* bekommt):

```
#include <iostream.h>
void main()
{
    cout << "Guten Tag!";
}
```
- // Schrägstriche leiten einen Kommentar ein;
- # Doppelkreuz leitet Compiler-Direktive ein;
- *include* Einfügen der angegebenen Datei in die Quelldatei;
- *iostream.h* enthält Vereinbarungen für die Ein- u. Ausgabe;
- *main()* Funktion, deren Anweisungen nach dem Programmstart abgearbeitet werden;
- *void* legt fest, daß die Funktion keine Rückgabewerte (Funktionswerte) liefert;
- Es gibt 48 Schlüsselwörter (z. B. *void, const, double, int, char*), diese müssen klein geschrieben werden.
- Bezeichner (Namen) können mit Groß- und Kleinbuchstaben geschrieben werden. Man beachte aber, daß dann z. B. *rad* ≠ *Rad*.
- Es wird empfohlen, Konstanten vollständig mit großen Buchstaben zu schreiben.

- **Start einer Entwicklungsumgebung, Eingabe des Programms in den Computer, Korrektur und Test**
- **Dokumentation, Wartung und Pflege des Programms**

9.2 Die Entwicklungsumgebung von Visual C++

Start: Ordner *Programmierung*, *MS Visual Studio 6.0*, *MS Visual C++ 6.0* wählen, liefert ein Anwendungsfenster mit mehreren Fenstern (z. B. *Arbeitsbereich*, *Ausgabe*).

Alle Dateien, die für die Lauffähigkeit eines Programms notwendig sind (Arbeitsbereichsdatei, Make-, Objekt-, Cpp-, Exe-, ...) werden in einem **Projekt** zusammengefaßt. Mehrere Projekte kann man zu einem **Arbeitsbereich** zusammenfassen. Alle Dateien eines Projektes haben denselben Namen mit verschiedenen Extensionen (*.dsw, *.dsp, *.obj, *.cpp, *.exe).

Bei der Anlage eines neuen Projektes wird automatisch ein neuer Arbeitsbereich gleichen Namens angelegt.

Aufgabe 9.2.1:

- Visual C++ starten** und wie folgt ein neues **Projekt Kugel anlegen**:
 - *Datei, Neu* wählen
 - Im Fenster *Neu* die Registerkarte *Projekte* wählen
 - Projektname: *Kugel* eintragen
 - Pfad: Pfad zum Zielordner eintragen
 - Plattform: *Win32 Konsolenanwendung* anklicken
- C++-Quelltext schreiben**: *Datei, Neu, C++-Quellcodedatei* wählen, Dateiname *Kugel.cpp* eintragen, *OK* und Quelltext wie oben erfassen und speichern,
- Kompilieren und Korrigieren**: *Erstellen, Kompilieren von Kugel.cpp* wählen → Quelltext wird in Maschinensprache übersetzt und im Ausgabefenster links unten kann man Kompilermittteilungen verfolgen, bei Fehlerangaben diese korrigieren und erneut kompilieren,
- Exe erstellen**: *Erstellen, Kugel.exe erstellen* wählen,
- Ausführen**: *Erstellen, Ausführen von Kugel.exe* wählen (oder einfach auf das Ausrufezeichen klicken) → in einem DOS-Fenster werden nun die Aus- und Eingaben vorgenommen.

Aufgabe 9.2.2:

- Projekt *Kugel* schließen und *Visual C++* beenden, im Arbeitsplatzfenster in den eigenen Ordner *Kugel* wechseln und über die erstellten Dateien informieren:

*.dsp	Makedatei (Developer Studio Projekt)
*.dsw	Arbeitsbereichsdatei (Developer Studio Workspace)
*.ncb	No Compile Browser
*.opt	Arbeitsbereichsoptionsdatei
*.plg	Erstellungsprotokoll
- In den Ordner *Debug* wechseln und über die Dateien informieren:

*.ilk	Linkinformationen
*.obj	Objektdatei
*.pch	precompiled header
*.pdb	Debug Informationen

Aufgabe 9.2.3:

- Erstellen eines Struktogramms und eines C++-Programms, das das Vertauschen der Werte zweier Variablen realisiert.
- Erstellen eines Projektes *Tausch*, Quelltext *Tausch.cpp* eingeben, kompilieren und ausführen.