

Technische Probleme lösen mit C/C++

Von der Analyse bis zur Dokumentation

DAS INHALTS- VERZEICHNIS

» Hier geht's
direkt
zum Buch

Inhalt

Vorwort des Herausgebers	5
Vorwort der Autoren	7
Einleitung	15
1 Systematik der Problemlösung	19
1.1 Phasen der Programmentwicklung	19
1.2 Software-Lebenszyklus	21
1.3 Software-Entwicklungsverfahren	23
2 Erste Gehversuche mit C/C++	28
2.1 Warum gerade C/C++?	28
2.2 Compiler und Interpreter	30
2.3 Übersetzen eines C/C++-Programms	32
2.4 Programmstart	33
3 Die Entwicklungsumgebung Visual C++	34
3.1 Installation von VC++	34
3.2 Starten von VC++	36
3.3 Erstellen eines neuen Projektes	38
3.3.1 Win32-Projekte	39
3.3.1.1 Variante 1 - VC++ leistet Vorarbeit	40
3.3.1.2 Variante 2 - leeres Projekt	41
3.3.2 CLR-Projekte	44
3.4 Übersetzen eines eigenen Programms	46
3.5 Ausführen eines eigenen Programms	49
3.6 Paradigmen der Projektorganisation	49
3.7 Ein erstes Programm in C/C++	51

4	Grundlegende Sprach- und Steuerungselemente	54
4.1	Kommentare	54
4.2	Datentypen und Variablen	55
4.2.1	Variablenamen	56
4.2.2	Ganzzahlige Variablen	56
4.2.3	Fließkommazahlen	58
4.2.4	Zeichen	59
4.2.5	Felder	60
4.2.5.1	Eindimensionale Felder	60
4.2.5.2	Mehrdimensionale Felder	61
4.2.5.3	Zugriff auf die Elemente eines Feldes	63
4.2.5.4	Startwertzuweisung für ein- und mehrdimensionale Arrays	65
4.2.6	Zeichenketten	67
4.3	Konstanten	68
4.4	Operatoren	69
4.4.1	Vorzeichenoperatoren	69
4.4.2	Arithmetische Operatoren	69
4.4.2.1	Addition +	69
4.4.2.2	Subtraktion -	69
4.4.2.3	Multiplikation *	70
4.4.2.4	Division /	70
4.4.2.5	Modulo %	70
4.4.2.6	Zuweisung =	70
4.4.2.7	Kombinierte Zuweisungen	71
4.4.2.8	Inkrementierung ++	71
4.4.2.9	Dekrementierung -	72
4.4.3	Vergleichsoperatoren	72
4.4.3.1	Gleichheit ==	72
4.4.3.2	Ungleichheit !=	72
4.4.3.3	Kleiner <	73
4.4.3.4	Größer >	73
4.4.3.5	Kleiner gleich <=	73
4.4.3.6	Größer gleich >=	74
4.4.4	Logische Operatoren	74
4.4.4.1	Logisches NICHT !	74
4.4.4.2	Logisches UND &&	74
4.4.4.3	Logisches ODER	74
4.4.5	Typumwandlungsoperator	75
4.4.6	Speicherberechnungsoperator	75
4.4.7	Bedingungsoperator	76
4.4.8	Indizierungsoperator	77
4.4.9	Klammerungsoperator	77
4.5	Anweisungen und Blöcke	79
4.6	Alternationen	79
4.6.1	Einfache Abfragen (if - else)	79

4.6.2	Mehrfachabfragen (else - if)	80
4.6.3	Die switch-case-Anweisung	81
4.7	Iterationen	83
4.7.1	Zählergesteuerte Schleifen (for)	83
4.7.2	Kopfgesteuerte Schleifen (while)	87
4.7.3	Fußgesteuerte Schleifen (do - while)	88
4.7.4	Schleifenabbruch (continue)	89
4.7.5	Schleifenabbruch (break)	90
4.7.6	Schleifenumwandlungen	92
4.8	Funktionen	92
4.8.1	Formaler Aufbau einer Funktion	93
4.8.1.1	Der Funktionskopf	94
4.8.1.2	Der Funktionsrumpf	95
4.8.2	Datentyp und Deklaration einer Funktion - Prototyping	96
4.8.3	Das Prinzip der Parameterübergabe	101
4.8.3.1	Aufrufverfahren call by value	101
4.8.3.2	Aufrufverfahren call by reference	103
4.8.3.3	Adressoperator, Zeiger und Dereferenzierung	106
4.8.4	Regeln für ein erfolgreiches Prototyping	107
4.8.5	Die exit()-Funktion	108
4.8.6	Rekursive Funktionen	108
4.9	Ein- und Ausgabe	111
4.9.1	Formatierte Eingabe mit scanf()	111
4.9.2	Formatierte Ausgabe mit printf()	112
4.9.3	Arbeiten mit Dateien	113
4.9.3.1	Öffnen der Datei	114
4.9.3.2	Verarbeiten der Datensätze	114
4.9.3.3	Schließen der Datei	115
4.9.3.4	stdio.h	115
4.9.3.5	fflush() und stdin	117
5	Strukturierte Programmierung	118
5.1	Problemstellung	119
5.2	Problemanalyse	120
5.3	Struktogramm nach Nassi-Shneiderman	123
5.3.1	Sequenz	125
5.3.2	Alternation	127
5.3.3	Verschachtelung	128
5.3.4	Verzweigung	129
5.3.5	Schleifen	131
5.3.5.1	Zählergesteuerte Schleife	131
5.3.5.2	Kopfgesteuerte Schleife	135
5.3.5.3	Fußgesteuerte Schleifen	137
5.3.5.4	Endlosschleifen	138
5.3.5.5	Kriterien zur Schleifenauswahl	138
5.3.6	Programm- oder Funktionsaufruf	138

5.3.7	Aussprung	139
5.3.8	Rechnergestützte Erstellung von Struktogrammen	140
5.3.8.1	StruktEd	140
5.3.8.2	hus-Struktogrammer	147
5.4	Flussdiagramm nach DIN 66001	155
5.5	Programmerstellung	157
5.6	Programmtest	157
5.7	Programmlauf	158
5.8	Dokumentation nach DIN 66230	159
5.8.1	Funktion und Aufbau des Programms	159
5.8.2	Programmkenndaten	160
5.8.3	Betrieb des Programms	161
5.8.4	Ergänzungen	161
5.9	Aspekte des Qualitätsmanagements EN-ISO 9000	162
5.10	Algorithmus – was ist das?	163
5.11	EVA-Prinzip	169
5.12	Programmierung von Formelwerken	170
6	Lösung einfacher Probleme	175
6.1	Umrechnung von Temperatursystemen	175
6.2	Flächenberechnung geradlinig begrenzter Flächen (Polygone)	181
6.2.1	Erste Problemvariation: Berechnung der Schwerpunkt- koordinaten $S(x_S; y_S)$ von polygonförmig begrenzten Flächen	188
6.2.2	Zweite Problemvariation: Suche nach einem „günstigen“ Treffpunkt	189
6.2.3	Eine Projektidee: Wohnflächenberechnung	190
6.3	Berechnung einer Brückenkonstruktion	191
6.4	Schaltjahrüberprüfung	195
6.5	Ein Problem aus der Energiewirtschaft	201
6.6	Logarithmische Achsenteilung	211
6.7	Berechnung der Kreiszahl π	218
6.7.1	Berechnung nach Archimedes (287 – 212 v. Chr.)	219
6.7.2	Berechnung mit der Monte-Carlo-Methode	221
6.7.3	π in C/C++-Programmen	226
6.8	Primzahlen – Sieb des Eratosthenes	227
6.9	Lottozahlen	232
7	Objektorientierte Programmierung (OOP)	235
7.1	Modellbildung mittels Abstraktion	235
7.2	Klassen und Objekte	236
7.3	Attribute und Methoden einer Klasse	239
7.4	Bruchrechnung mit OOP	240
7.5	Vererbung	249
7.6	Strings	255
7.7	Typumwandlungen	257
7.8	Strukturierte Programmierung vs. OOP	260

8	Lösung fortgeschrittener Probleme	262
8.1	Grafische Darstellung funktionaler Abhängigkeiten	262
8.1.1	Welt- und Screenkoordinaten	264
8.1.2	Koordinatentransformationen	266
8.1.3	Darstellung der Sinusfunktion	272
8.1.4	Darstellung quadratischer Parabeln	276
8.1.5	Spannungsteilerkennlinien	279
8.2	Lösung technisch-wissenschaftlicher Probleme	281
8.2.1	Widerstandsreihen E6 bis E96	281
8.2.2	Farbcodierung von Widerständen nach DIN 41429	284
8.2.3	Fourier-Synthese periodischer empirischer Funktionen	287
8.2.4	Fourier-Analyse empirischer Funktionen	295
8.3	Nullstellenbestimmung von Funktionen	300
8.3.1	Inkrementverfahren und Intervallhalbierung	300
8.3.2	Die regula falsi	305
8.3.3	Das Newton-Verfahren	307
8.4	Numerische Integration	310
8.4.1	Riemannsches Unter- und Obersummen	310
8.4.2	Trapezregel	314
8.4.3	Simpsonsche Regel	319
8.4.4	Effektivwertberechnungen	324
8.4.5	Volumenberechnung	326
8.5	Einbindung eigener Klassen	334
8.5.1	Das „Platinenproblem“ als objektorientierte Konsolenanwendung	334
8.5.2	Das „Platinenproblem“ in der Erweiterung mit grafischer Benutzeroberfläche	339
9	Lösung komplexer Probleme	343
9.1	Kurvendiskussion und Funktionsplotter am Beispiel ganzrationaler Funktionen bis 3. Ordnung	343
9.2	Ausgleichsrechnung – Bestimmung der „besten“ Geraden in einer Messreihe	346
9.3	Digitaltechnik	356
9.4	Sortierverfahren	370
9.4.1	Bubble-Sort	370
9.4.2	Ripple-Sort	374
9.4.3	Sortieren alphanumerischer Daten mit Ripple-Sort	377
9.4.4	Sortieren von Listen nach vorgegebenen Kriterien	380
9.4.5	Benchmarkprogramm zur Beurteilung der Leistungsfähigkeit von Sortierprogrammen	386
9.4.5.1	Die Sortierprogramme Bubble-, Linear-, Ripple-, Quick- und Shell-Sort im Vergleich	386
9.4.5.2	Zeiterfassung in der Programmiersprache C/C++	387
9.4.5.3	Das Benchmarkprogramm	391
9.4.5.4	Interpretation der Vergleichsergebnisse	396
9.4.5.5	Die O-Notation	397
9.5	40 Jahre EDV mit dem PC – ein Blick zurück und nach vorn	398

10 Tabellen und Übersichten	401
10.1 Datentypen und ihre Wertebereiche	401
10.2 Vergleich der Symbole nach DIN 66 001 und der Nassi-Shneiderman- Darstellung	402
10.3 Schlüsselwörter ANSI C	403
10.4 Erweiterte Schlüsselwörter C++	405
10.5 ASCII-Tabelle	408
10.6 Standardfunktionen und ihre Zuordnung zu den Header-Dateien (include)	410
 Literatur	 414
 Index	 415