
SPIIS TREŚCI

O autorze	19
Przedmowa	21
Rozdział 1. Zaczynamy	23
Skąd C?	23
Dlaczego C?	24
Cechy użytkowe	25
Efektywność	25
Przenośność	25
Moc i elastyczność	26
Ukierunkowanie na programistę	26
Słabe strony	26
Dokąd zmierza C?	27
Co robią komputery?	28
Języki wysokiego poziomu i kompilatory	29
Standardy języka	30
Standard ANSI/ISO C	31
Standard C99	31
Standard C11	32
Korzystanie z C — siedem kroków	33
Krok 1. Określenie celów programu	33
Krok 2. Projektowanie programu	34
Krok 3. Pisanie kodu	34
Krok 4. Kompilacja	35
Krok 5. Uruchomienie programu	35
Krok 6. Testowanie i usuwanie błędów	35
Krok 7. Pielęgnowanie i modyfikowanie programu	36
Komentarz	36
Mechanika programowania	37
Pliki kodu obiektowego, pliki wykonywalne i biblioteki	38
UNIX	39
GNU Compiler Collection i LLVM	41

Linux	42
Kompilatory dla komputerów PC	43
Zintegrowane środowiska programistyczne (Windows)	43
Opcja podwójna — Windows/Linux	45
Jezyk C a komputery Mac	45
Jak zorganizowano tę książkę	46
Konwencje zapisu	46
Czcionka	47
Tekst na ekranie	47
Informacje dodatkowe	48
Podsumowanie rozdziału	49
Pytania sprawdzające	49
Ćwiczenie	49
Rozdział 2. Wstęp do C	51
Prosty przykład języka C	51
Objaśnienie	53
Podejście 1. Szybkie streszczenie	53
Podejście 2. Szczegóły	55
Budowa prostego programu	64
Co zrobić, aby Twój program był czytelny?	65
Kolejny krok	66
Dokumentacja	66
Wielokrotne deklaracje	67
Mnożenie	67
Wyświetlanie wielu wartości	67
Wiele funkcji	68
Usuwanie błędów	69
Błędy składniowe	70
Błędy semantyczne	71
Stan programu	72
Słowa kluczowe	73
Kluczowe zagadnienia	74
Podsumowanie rozdziału	74
Pytania sprawdzające	75
Ćwiczenia	76
Rozdział 3. Dane w C	79
Program przykładowy	79
Co nowego?	81
Zmienne i stałe	82
Słowa kluczowe typów danych	83
Typy całkowite a typy zmiennoprzecinkowe	84
Liczba całkowita	85
Liczba zmiennoprzecinkowa	85

Typy danych w C	86
Typ int	86
Inne typy całkowite	91
Korzystanie ze znaków — typ char	96
Typ _Bool	102
Typy przenośne	102
Typy float, double, long double	105
Typy zespolone i urojone	110
Inne typy	111
Rozmiary typów	113
Korzystanie z typów danych	114
Uwaga na argumenty	115
Jeszcze jeden przykład	117
Co się dzieje?	117
Bufor wyjścia	118
Kluczowe zagadnienia	119
Podsumowanie rozdziału	119
Pytania sprawdzające	120
Ćwiczenia	122
Rozdział 4. Łańcuchy znakowe i formatowane wejście-wyjście	125
Na początek... program	126
Łańcuchy znakowe. Wprowadzenie	127
Tablice typu char i znak zerowy	127
Korzystanie z łańcuchów	128
Funkcja strlen()	130
Stałe i preprocesor C	132
Modyfikator const	135
Stałe standardowe	136
Poznać i wykorzystać printf() i scanf()	138
Funkcja printf()	138
Korzystanie z printf()	139
Modyfikatory specyfikatorów konwersji dla printf()	141
Znaczenie konwersji	147
Korzystanie z funkcji scanf()	154
Modyfikator * w funkcjach printf() i scanf()	160
Praktyczne wskazówki	161
Kluczowe zagadnienia	163
Podsumowanie rozdziału	164
Pytania sprawdzające	164
Ćwiczenia	167
Rozdział 5. Operatory, wyrażenia i instrukcje	169
Wstęp do pętli	170
Podstawowe operatory	172
Operator przypisania: =	172
Operator dodawania: +	175

Operator odejmowania: -	176
Operatory znaku: - i +	176
Operator mnożenia: *	177
Operator dzielenia: /	179
Priorytet operatorów	180
Priorytet i kolejność obliczeń	182
Niektóre inne operatory	183
Operator sizeof i typ size_t	183
Operator modulo: %	184
Operatory inkrementacji i dekrementacji: ++ i --	186
Dekrementacja --	190
Priorytet	191
Nie próbuj być zbyt sprytny	191
Wyrażenia i instrukcje	193
Wyrażenia	193
Instrukcje	194
Instrukcje złożone (bloki)	197
Konwersje typów	199
Operator rzutowania	202
Funkcje z argumentami	203
Przykładowy program	206
Zagadnienia kluczowe	207
Podsumowanie rozdziału	208
Pytania sprawdzające	209
Ćwiczenia	212
Rozdział 6. Instrukcje sterujące C. Pętle	215
Wracamy do pętli while	216
Komentarz	217
Pętla odczytująca w stylu C	219
Instrukcja while	219
Zakończenie pętli while	220
Kiedy kończy się pętla?	220
while jako pętla z warunkiem wejścia	221
Wskazówki dotyczące składni	222
Co jest większe? Korzystanie z operatorów i wyrażen relacyjnych	223
Czym jest prawda?	225
Co jeszcze jest prawdą?	226
Problemy z prawdą	227
Nowy typ _Bool	229
Priorytet operatorów relacyjnych	231
Pętle nieokreślone i pętle liczące	232
Pętla for	234
Elastyczność pętli for	235
Inne operatory przypisania: +=, -=, *=, /=, %=	239

Operator przecinkowy: ,	241
Zenon z Elei kontra pętla for	244
Pętla z warunkiem wyjścia — do while	245
Której pętli użyć?	248
Pętle zagnieżdżone	249
Omówienie	250
Inny wariant	250
Tablice	251
Współpraca tablicy i pętli for	252
Przykład wykorzystujący pętlę i wartość zwracaną przez funkcję	254
Omówienie programu	257
Korzystanie z funkcji zwracających wartości	258
Zagadnienia kluczowe	258
Podsumowanie rozdziału	259
Pytania sprawdzające	260
Ćwiczenia	264
Rozdział 7. Instrukcje sterujące C. Rozgałęzienia i skoki	269
Instrukcja if	270
Dodajemy else	272
Kolejny przykład: funkcje getchar() i putchar()	273
Rodzina funkcji znakowych ctype.h	276
Wybór spośród wielu możliwości — else if	278
Łączenie else z if	281
Więcej o zagnieżdżonych instrukcjach if	283
Bądźmy logiczni	287
Zapis alternatywny — plik nagłówkowy iso646.h	289
Priorytet	289
Kolejność obliczeń	290
Zakresy	291
Program liczący słowa	292
Operator warunkowy: ?:	296
Dodatki do pętli — continue i break	298
Instrukcja continue	298
Instrukcja break	301
Wybór spośród wielu możliwości — switch i break	304
Korzystanie z instrukcji switch	305
Pobieranie tylko pierwszego znaku w wierszu	307
Etykiety wielokrotne	308
Switch a if else	309
Instrukcja goto	311
Unikanie goto	311
Kluczowe zagadnienia	314
Podsumowanie rozdziału	315
Pytania sprawdzające	316
Ćwiczenia	319

Rozdział 8. Znakowe wejście-wyjście i przekierowywanie	323
Jednoznakowe we-wy — getchar() i putchar()	324
Bufory	325
Kończenie danych wprowadzanych z klawiatury	327
Pliki, strumienie i dane wprowadzane z klawiatury	327
Koniec pliku	329
Przekierowywanie a pliki	332
Przekierowywanie w systemach UNIX, Linux i Windows	332
Tworzenie przyjaźniejszego interfejsu użytkownika	337
Współpraca z buforowanym wejściem	337
Łączenie wejścia liczbowego i znakowego	340
Sprawdzanie poprawności danych wejściowych	343
Analiza programu	347
Strumienie wejściowe a liczby	348
Menu	349
Zadania	349
W kierunku sprawnego działania	350
Łączenie danych znakowych i numerycznych	352
Zagadnienia kluczowe	355
Podsumowanie rozdziału	356
Pytania sprawdzające	356
Ćwiczenia	357
Rozdział 9. Funkcje	361
Przypomnienie	361
Tworzenie i wykorzystanie prostej funkcji	363
Analiza programu	363
Argumenty funkcji	366
Definiowanie funkcji pobierającej argument — argumenty formalne	368
Prototyp funkcji pobierającej argumenty	369
Wywoływanie funkcji pobierającej argumenty — argumenty faktyczne	369
Punkt widzenia czarnej skrzynki	370
Zwracanie wartości z wykorzystaniem instrukcji return	371
Typy funkcji	373
Prototypy ANSI C	375
Problem	375
ANSI C na ratunek!	376
Brak argumentów a argumenty nieokreślone	377
Potęga prototypów	378
Rekurencja	379
Rekurencja bez tajemnic	379
Podstawy rekurencji	380
Rekurencja końcowa	382
Rekurencja i odwracanie kolejności działań	384
Za i przeciw rekurencji	386

Kompilowanie programów zawierających więcej niż jedną funkcję	387
Unix	387
Linux	388
DOS (kompilatory wiersza poleceń)	388
Środowiska IDE dla Windows i OS X	388
Korzystanie z plików nagłówkowych	388
Uzyskiwanie adresów: operator &	392
Modyfikacja zmiennych w funkcji wywołującej	394
Wskaźniki: pierwsze spojrzenie	396
Operator dereferencji: *	396
Deklarowanie wskaźników	396
Wykorzystanie wskaźników do komunikacji pomiędzy funkcjami	398
Kluczowe zagadnienia	402
Podsumowanie rozdziału	403
Pytania sprawdzające	403
Ćwiczenia	404
Rozdział 10. Tablice i wskaźniki	407
Tablice	407
Inicjalizacja	408
Oznaczona inicjalizacja (C99)	412
Przypisywanie wartości do tablic	414
Zakres tablic	414
Określanie rozmiaru tablicy	416
Tablice wielowymiarowe	417
Inicjalizacja tablicy dwuwymiarowej	420
Więcej wymiarów	421
Wskaźniki do tablic	422
Funkcje, tablice i wskaźniki	425
Korzystanie z parametrów wskaźnikowych	428
Komentarz — wskaźniki i tablice	430
Działania na wskaźnikach	430
Ochrona zawartości tablicy	435
Zastosowanie słowa kluczowego const w parametrach formalnych	436
Więcej o const	437
Wskaźniki a tablice wielowymiarowe	439
Wskaźniki do tablic wielowymiarowych	442
Zgodność wskaźników	444
Funkcje a tablice wielowymiarowe	446
Tablice o zmiennym rozmiarze (VLA, ang. variable — length array)	449
Literały złożone	453
Zagadnienia kluczowe	456
Podsumowanie rozdziału	456
Pytania sprawdzające	458
Ćwiczenia	460

Rozdział 11. Łańcuchy znakowe i funkcje łańcuchowe	463
Reprezentacja łańcuchów i łańcuchowe wejście-wyjście	463
Definiowanie łańcuchów	464
Wskaźniki a łańcuchy	473
Wczytywanie łańcuchów	475
Tworzenie miejsca	475
Niesławna funkcja gets()	475
Alternatywy dla funkcji gets()	477
Funkcja scanf()	484
Wyświetlanie łańcuchów	486
Funkcja puts()	486
Funkcja fputs()	488
Funkcja printf()	488
Zrób to sam	489
Funkcje łańcuchowe	491
Funkcja strlen()	492
Funkcja strcat()	493
Funkcja strncat()	495
Funkcja strcmp()	496
Funkcje strcmp() i strncmp()	503
Funkcja sprintf()	508
Inne funkcje łańcuchowe	509
Przykład użycia. Sortowanie łańcuchów	512
Sortowanie wskaźników zamiast łańcuchów	513
Algorytm sortowania przez selekcję	514
Łańcuchy a funkcje znakowe z rodziny ctype.h	515
Argumenty wiersza poleceń	517
Argumenty wiersza poleceń w środowiskach zintegrowanych	519
Argumenty linii poleceń w systemie Macintosh	520
Konwersja łańcuchów do liczb	520
Zagadnienia kluczowe	523
Podsumowanie rozdziału	524
Pytania sprawdzające	525
Ćwiczenia	528
Rozdział 12. Klasy zmiennej, łączność i zarządzanie pamięcią	531
Klasy zmiennych	532
Zasięg zmiennej	533
Łączność zmiennej	535
Czas trwania zmiennej	537
Zmienne automatyczne	538
Zmienne rejestrowe	543
Zmienne statyczne o zasięgu blokowym	543
Zmienne statyczne o łączności zewnętrznej	545
Zmienne statyczne o łączności wewnętrznej	550
Programy wieloplikowe	551

Specyfikatory klasy zmiennych — podsumowanie	551
Klasy zmiennych a funkcje	554
Którą klasę wybrać?	555
Funkcje pseudolosowe i zmienne statyczne	555
Rzut kostką	559
Przydział pamięci. Funkcje malloc() i free()	563
Znaczenie funkcji free()	568
Funkcja calloc()	568
Dynamiczny przydział pamięci a tablice o zmiennym rozmiarze	569
Klasy zmiennych a dynamiczny przydział pamięci	570
Kwalifikatory typu ANSI C	572
Kwalifikator typu const	572
Kwalifikator typu volatile	575
Kwalifikator typu restrict	576
Kwalifikator _Atomic (C11)	577
Stare słowa kluczowe w nowych miejscach	578
Kluczowe zagadnienia	579
Podsumowanie rozdziału	579
Pytania sprawdzające	581
Ćwiczenia	582
Rozdział 13. Obsługa plików	587
Wymiana informacji z plikami	587
Czym jest plik?	588
Poziomy wejścia-wyjścia	590
Pliki standardowe	590
Standardowe wejście-wyjście	591
Sprawdzanie argumentów wiersza poleceń	592
Funkcja fopen()	593
Funkcje getc() i putc()	595
Znak końca pliku EOF (ang. end of file)	595
Funkcja fclose()	596
Wskaźniki do plików standardowych	597
Niewyszukany program kompresujący pliki	597
Plikowe wejście-wyjście — fprintf(), fscanf(), fgets() i fputs()	599
Funkcje fprintf() i fscanf()	599
Funkcje fgets() i fputs()	601
Przygody z dostępem swobodnym — fseek() i ftell()	602
Jak działają funkcje fseek() i ftell()?	603
Tryb binarny a tryb tekstowy	604
Przenośność	605
Funkcje fgetpos() i fsetpos()	606
Za kulisami standardowego wejścia-wyjścia	606
Inne standardowe funkcje wejścia-wyjścia	607
Funkcja int ungetc()	608
Funkcja int fflush()	608
Funkcja int setvbuf()	608

Binarne wejście-wyjście: fread() i fwrite()	609
Funkcja size_t fwrite()	611
Funkcja size_t fread()	611
Funkcje int feof(FILE *fp) oraz int ferror(FILE *fp)	612
Przykład	612
Dostęp swobodny w binarnym wejściu-wyjściu	615
Zagadnienia kluczowe	617
Podsumowanie rozdziału	618
Pytania sprawdzające	619
Ćwiczenia	621
Rozdział 14. Struktury i inne formy danych	625
Przykładowy problem. Tworzenie spisu książek	626
Deklaracja struktury	627
Definiowanie zmiennej strukturalnej	628
Inicjalizacja struktury	630
Odwołania do składników struktury	630
Inicjatory oznaczone struktur	631
Tablice struktur	632
Deklarowanie tablicy struktur	634
Wskazywanie składników tablicy struktur	634
Szczegóły programu	635
Struktury zagnieżdżone	636
Wskaźniki do struktur	638
Deklaracja i inicjalizacja wskaźnika do struktury	639
Dostęp do składników za pomocą wskaźnika	640
Struktury a funkcje	641
Przekazywanie składników struktur	641
Korzystanie z adresu struktury	642
Przekazywanie struktury jako argumentu	643
Więcej o nowym, ulepszonym statusie struktury	644
Struktury czy wskaźniki do struktur?	648
Tablice znakowe lub wskaźniki do znaków w strukturze	649
Struktury, wskaźniki i funkcja malloc()	650
Literały złożone i struktury (C99)	652
Elastyczne składniki tablicowe (C99)	654
Struktury anonimowe (C11)	657
Funkcje korzystające z tablic struktur	657
Zapisywanie zawartości struktury w pliku	659
Przykład zapisu struktury	660
Omówienie programu	663
Struktury. Co dalej?	664
Unie. Szybkie spojrzenie	665
Wykorzystywanie unii	666
Unie anonimowe (C11)	667

Typy wyliczeniowe	669
Stałe enum	670
Wartości domyślne	670
Przypisywane wartości	670
Użycie enum	670
Współdzielona przestrzeń nazw	672
typedef: szybkie spojrzenie	673
Udziwnione deklaracje	675
Funkcje a wskaźniki	677
Kluczowe zagadnienia	684
Podsumowanie rozdziału	685
Pytania sprawdzające	686
Ćwiczenia	689
Rozdział 15. Manipulowanie bitami	693
Liczby binarne, bity i bajty	694
Binarne liczby całkowite	694
Liczby całkowite ze znakiem	695
Binarne liczby zmiennoprzecinkowe	696
Inne systemy liczbowe	697
System ósemkowy	697
System szesnastkowy	698
Operatory bitowe	698
Bitowe operatory logiczne	699
Zastosowanie. Maski	701
Zastosowanie. Ustawianie bitów (włączanie bitów)	702
Zastosowanie. Zerowanie bitów (wyłączanie bitów)	702
Zastosowanie. Odwracanie bitów	703
Zastosowanie. Sprawdzenie wartości bitu	703
Bitowe operatory przesunięcia	704
Przykład	706
Kolejny przykład	708
Pola bitowe	710
Przykład	711
Pola bitowe a operatory bitowe	715
Mechanizmy wyrównania danych (C11)	722
Kluczowe zagadnienia	724
Podsumowanie rozdziału	725
Pytania sprawdzające	726
Ćwiczenia	727
Rozdział 16. Preprocesor i biblioteka C	731
Pierwsze kroki w translacji programu	732
Stałe symboliczne. #define	733
Tokeny	737
Przeddefiniowywanie stałych	737

#define i argumenty	738
Argumenty makr w łańcuchach	741
Łącznik preprocesora. Operator ##	742
Makra o zmiennej liczbie argumentów: ... i __VA_ARGS__	743
Makro czy funkcja?	744
Dołączanie plików. #include	746
Pliki nagłówkowe. Przykład	747
Zastosowania plików nagłówkowych	749
Inne dyrektywy	750
Dyrektywa #undef	750
Zdefiniowany. Z perspektywy preprocesora C	751
Kompilacja warunkowa	751
Makra predefiniowane	756
#line i #error	757
#pragma	758
Słowo kluczowe _Generic (C11)	759
Funkcje wplatane (C99)	761
Funkcje bezpowrotne (C11)	764
Biblioteka języka C	764
Uzyskiwanie dostępu do biblioteki C	764
Korzystanie z opisów funkcji	765
Biblioteka funkcji matematycznych	767
Odrobina trygonometrii	768
Warianty typów zmiennoprzecinkowych	770
Biblioteka tmath.h (C99)	771
Biblioteka narzędzi ogólnego użytku	772
Funkcje exit() i atexit()	773
Funkcja qsort()	775
Biblioteka assert.h	780
Stosowanie asercji	780
_Static_assert (C11)	781
Funkcje memcpy() i memmove() z biblioteki string.h	782
Zmienna liczba argumentów. stdarg.h	785
Zagadnienie kluczowe	787
Podsumowanie rozdziału	788
Pytania sprawdzające	788
Ćwiczenia	790
Rozdział 17. Zaawansowana reprezentacja danych	793
Poznajemy reprezentację danych	794
Listy łączone	797
Korzystanie z listy łączonej	801
Refleksje	805
Abstrakcyjne typy danych (ATD)	806
Więcej abstrakcji	807
Budowanie interfejsu	808

Korzystanie z interfejsu	813
Implementacja interfejsu	815
Kolejki	822
Definicja kolejki jako abstrakcyjnego typu danych	822
Definicja interfejsu	823
Implementacja reprezentacji danych	824
Testowanie kolejki	832
Symulowanie za pomocą kolejki	834
Lista łączona czy tablica?	840
Drzewa binarne	844
Drzewo binarne jako ATD	846
Interfejs drzewa binarnego	846
Implementacja drzewa binarnego	849
Testowanie drzewa	863
Uwagi o drzewach	868
Co dalej?	869
Zagadnienia kluczowe	870
Podsumowanie rozdziału	871
Pytania sprawdzające	871
Ćwiczenia	872
Dodatek A. Odpowiedzi na pytania sprawdzające	875
Dodatek B. Dokumentacja	915
I. Lektura uzupełniająca	915
II. Operatory w języku C	919
III. Podstawowe typy i klasy zmiennych	925
IV. Wyrażenia, instrukcje i przepływ sterowania w programie	930
V. Standardowa biblioteka ANSI C oraz rozszerzenia standardu C99 i C11	937
VI. Rozszerzone typy całkowite	987
VII. Obsługa rozszerzonych zbiorów znaków	991
VIII. Efektywniejsze obliczenia numeryczne w C99 i C11	997
IX. Różnice między C a C++	1006
Skorowidz	1013