

Spis treści

O autorze	15
Podziękowania od autora	17
Wprowadzenie	19
CZĘŚĆ I: PODSTAWY PROGRAMOWANIA W C++	25
ROZDZIAŁ 1: Pierwszy program w C++	27
Pojęcia dotyczące języka C++	28
Instalacja Code::Blocks	29
Windows	29
Ubuntu Linux	32
Macintosh	34
Tworzenie pierwszego programu w C++	37
Tworzenie projektu	38
Wprowadzanie kodu w języku C++	39
Korzystanie ze ściągi	41
Budowanie programu	42
Uruchamianie programu	43
Przeglądanie programu z komentarzami	44
Analiza struktury programów w C++	45
Opatrywanie kodu źródłowego komentarzami	45
Tworzenie programów z instrukcji C++	46
Deklaracja zmiennych	47
Generowanie wyjścia programu	48
Obliczanie wyrażeń	48
Przechowywanie wyników wyrażenia	48
Pozostała część programu Conversion	49

ROZDZIAŁ 2:	Stałe deklarowanie zmiennych	51
	Deklaracja zmiennych	52
	Deklarowanie różnych typów zmiennych	52
	Przegląd ograniczeń liczb całkowitych w C++	53
	Rozwiązywanie problemu zaokrągleń	54
	Ograniczenia liczb zmiennoprzecinkowych	55
	Deklarowanie typów zmiennych	57
	Typy stałych	58
	Zakres typów liczbowych	59
	Znaki specjalne	60
	Szerokie ładunki na autostradzie typu Char	61
	Czy te obliczenia są logiczne?	62
	Wyrażenia mieszane	63
	Deklaracje automatyczne	64
ROZDZIAŁ 3:	Operacje matematyczne	67
	Wykonywanie prostych obliczeń binarnych	68
	Dekomponowanie wyrażeń	69
	Określanie kolejności operacji	69
	Wykonywanie operacji jednoargumentowych	70
	Korzystanie z operatorów przypisania	72
ROZDZIAŁ 4:	Operacje logiczne	75
	Po co wykonujemy operacje logiczne?	76
	Korzystanie z prostych operatorów logicznych	76
	Przechowywanie wartości logicznych	77
	Używanie logicznych zmiennych typu int	79
	Zawiłości wykonywania operacji logicznych na zmiennych zmiennoprzecinkowych	79
	Wyrażanie liczb binarnych	81
	System liczb dziesiętnych	81
	Inne systemy liczbowe	82
	System liczb binarnych	82
	Wykonywanie bitowych operacji logicznych	84
	Operatory jednobitowe	84
	Używanie operatorów bitowych	86
	Prosty test	86

ROZDZIAŁ 5: Sterowanie przepływem programu89

Sterowanie przepływem programu za pomocą instrukcji rozgałęzienia 90

Wykonywanie pętli 92

 Wykonywanie pętli, gdy warunek jest prawdziwy 92

 Wykorzystanie autoinkrementacji i autodekrementacji 94

 Pętla for 95

 Unikanie budzącej strach pętli nieskończonej 98

 Pętla for zależna od zakresu 98

 Instrukcje zarządzania pętlami 99

Zagnieżdżanie instrukcji sterujących 102

Przełączanie na inny temat? 104

**CZĘŚĆ II: PROGRAMOWANIE
W C++ Z WYKORZYSTANIEM FUNKCJI 107**

ROZDZIAŁ 6: Tworzenie funkcji109

Pisanie i używanie funkcji 110

 Definiowanie funkcji 112

 Definiowanie funkcji sumujSekwencje() 112

 Wywoływanie funkcji sumujSekwencje() 113

 Dziel i rządź 113

Funkcje w szczegółach 113

 Funkcje proste 114

 Funkcje z argumentami 115

Przeciążanie funkcji 117

Definiowanie prototypów funkcji 119

Argumenty domyślne 120

Przekazywanie przez wartość i przekazywanie przez referencję 122

Typy przechowywania zmiennych 123

ROZDZIAŁ 7: Przechowywanie sekwencji w tablicach125

Szeregowanie argumentów w tablicach 126

 Korzystanie z tablic 127

 Inicjowanie tablic 130

 Sięganie poza zakres tablicy 131

 Pętle bazujące na zakresach 131

 Definiowanie i używanie tablic w tablicach 132

Korzystanie z tablic znaków 133

 Tworzenie tablicy znaków 133

 Tworzenie ciągu znaków 134

	Manipulowanie ciągami znaków	136
	Niektóre funkcje biblioteki standardowej	138
	Miejsce dla ciągów złożonych z szerokich znaków	139
ROZDZIAŁ 8:	Wskaźniki w C++	141
	Rozmiar zmiennej	141
	Czym jest adres?	143
	Operatory adresu	143
	Używanie wskaźników	145
	Używanie różnych typów wskaźników	146
	Przekazywanie wskaźników do funkcji	147
	Przekazywanie przez wartość	147
	Przekazywanie wartości wskaźnikowych	148
	Przekazywanie przez referencję	148
	Zmienne stałe	149
	Korzystanie ze sterty	151
	Ograniczony zasięg	151
	Analiza problemu zakresu ważności	152
	Użycie sterty	153
ROZDZIAŁ 9:	Drugie spojrzenie na wskaźniki w C++	155
	Definiowanie operacji na wskaźnikach	155
	Użycie wskaźników w tablicach	156
	Użycie operatorów do adresu tablicy	158
	Operacje wskaźnikowe na ciągach znaków	159
	Uzasadnienie użycia wskaźników	161
	Stosowanie operatorów do wskaźników innych typów niż char	161
	Porównanie wskaźnika z tablicą	162
	Wskaźnik null	164
	Deklaracja i użycie tablic wskaźników	165
	Wykorzystanie tablic ciągów znaków	166
	Dostęp do argumentów funkcji main()	168
ROZDZIAŁ 10:	Preprocesor C++	173
	Co to jest preprocesor?	173
	Dołączanie plików	174
	#Definiowanie	177
	A może by tak nie definiować?	180
	Wyliczeniowy typ danych	181
	Dołączanie warunkowe z użyciem instrukcji #if	183
	Obiekty zdefiniowane wewnętrznie	184
	Typedef	186

CZĘŚĆ III: WPROWADZENIE DO KLAS 187

ROZDZIAŁ 11: Programowanie obiektowe 189

Abstrakcyjne kuchenki mikrofalowe	189
Przygotowanie funkcjonalnego nachos	190
Przygotowywanie nachos w sposób obiektowy	191
Klasyfikacja kuchenki mikrofalowej	191
Po co klasyfikujemy przedmioty?	192

ROZDZIAŁ 12: Klasy w C++ 195

Wprowadzenie do klas	195
Kształt klasy	196
Dostęp do elementów klasy	197
Aktywacja obiektów	197
Symulowanie rzeczywistych obiektów	198
Po co zwracać sobie głowę funkcjami składowymi?	198
Dodawanie funkcji składowej	199
Wywoływanie funkcji składowych	200
Dostęp do pól z funkcji składowej	202
Ustalanie zakresu	203
Definiowanie funkcji składowych w klasie	204
Umieszczanie funkcji składowych poza klasą	206
Przeciążanie funkcji składowych	208

ROZDZIAŁ 13: Wskazywanie na obiekty 211

Deklarowanie tablic obiektów	211
Deklarowanie wskaźników do obiektów	212
Dereferencja wskaźnika obiektu	213
Wskazywanie w kierunku strzałek	214
Przekazywanie obiektów do funkcji	214
Wywoływanie funkcji z wartością obiektu	215
Wywoływanie funkcji ze wskaźnikiem na obiekt	216
Wywoływanie funkcji za pomocą operatora referencji	217
Po co przejmować się wskaźnikami i referencjami?	218
Zwracanie do sterty	219
Alokowanie obiektów ze sterty	220
Kiedy kompilator alokuje za Ciebie pamięć?	220
Listy	221
Wykonywanie innych operacji na listach	222
Wykorzystanie list w programie DaneListyJed	223
Promyk nadziei: lista kontenerów w bibliotece C++	226

ROZDZIAŁ 14:	Chronione elementy składowe klasy	227
	Ochrona składowych	227
	Dlaczego potrzebujesz składowych chronionych?	228
	Jak działają składowe chronione?	228
	Po co chronić składowe klasy?	230
	Ochrona wewnętrznego stanu klasy	230
	Używanie klasy z ograniczonym interfejsem	231
	Dostęp do elementów chronionych z zewnątrz	231
ROZDZIAŁ 15:	„Po co mnie tworzysz, skoro za chwilę chcesz mnie zniszczyć?”	235
	Tworzenie obiektów	236
	Korzystanie z konstruktorów	237
	Konstruowanie pojedynczego obiektu	237
	Konstruowanie wielu obiektów	238
	Konstruowanie dupleksu	239
	Dekonstrukcja obiektu	241
	Do czego służy destruktor?	241
	Praca z destruktorami	242
ROZDZIAŁ 16:	Argumenty przekazywane przez konstruktor	247
	Wyposażenie konstruktorów w argumenty	248
	Korzystanie z konstruktora	248
	Przeciążanie konstruktora	250
	Automatyczne konstruktory domyślne	253
	Konstruowanie składowych klasy	255
	Konstruowanie składowej typu złożonego	255
	Konstruowanie składowej, która jest stałą	261
	Rekonstrukcja kolejności konstruowania	261
	Kolejność konstrukcji obiektów lokalnych	262
	Obiekty statyczne są konstruowane tylko raz	262
	Wszystkie obiekty globalne są konstruowane przed funkcją main()	263
	Obiekty globalne są konstruowane bez określonej kolejności	264
	Składowe są konstruowane w kolejności, w jakiej zostały zadeklarowane	265
	Destruktry niszczą obiekty w kolejności odwrotnej do kolejności ich tworzenia	265
	Konstruowanie tablic	265
	Konstruktory jako forma konwersji	266

ROZDZIAŁ 17:	Konstruktory kopiujący i przenoszący	269
	Kopiowanie obiektu	269
	Do czego służy konstruktor kopiujący?	270
	Korzystanie z konstruktora kopiującego	270
	Automatyczny konstruktor kopiujący	272
	Kopiowanie głębokie i kopiowanie płytkie	273
	Obiekty tymczasowe	277
	Trwałe unikanie obiektów tymczasowych	279
	Konstruktor przenoszący	280
ROZDZIAŁ 18:	Składowe statyczne:	
	czy zmieknacz tkanin może pomóc?	283
	Definiowanie składowej statycznej	284
	Do czego służą składowe statyczne?	284
	Używanie składowych statycznych	284
	Odwoływanie się do składowych statycznych	285
	Przypadki użycia składowych statycznych	287
	Deklarowanie statycznych funkcji składowych	287
	O czym to w ogóle jest?	290
CZĘŚĆ IV: DZIEDZICZENIE		293
ROZDZIAŁ 19:	Dziedziczenie	295
	Czy potrzebuję dziedziczenia?	297
	Jak dziedziczy klasa?	297
	Używanie podklasy	299
	Konstruowanie podklasy	299
	Niszczanie podklasy	301
	Dziedziczenie konstruktorów	301
	Relacja MA	302
ROZDZIAŁ 20:	Wirtualne funkcje składowe. Czy są realne?	303
	Do czego jest potrzebny polimorfizm?	306
	Jak działa polimorfizm?	307
	Kiedy funkcja nie jest wirtualna?	308
	Wirtualne rozważania	310

ROZDZIAŁ 21:	Faktoryzacja klas	313
	Faktoryzacja	314
	Implementacja klas abstrakcyjnych	318
	Opisywanie koncepcji klasy abstrakcyjnej	318
	Tworzenie klasy konkretnej z klasy abstrakcyjnej	320
	Przekazywanie klas abstrakcyjnych	320

CZĘŚĆ V: BEZPIECZEŃSTWO 323

ROZDZIAŁ 22:	Czy zaakceptujesz nowy operator przypisania?	325
	Porównanie operatorów z funkcjami	326
	Wstawianie nowego operatora	327
	Tworzenie płytkich kopii to poważny problem	327
	Przeciążanie operatora przypisania	329
	Przeciążenie operatora indeksu	333
	Konstruktor i operator przenoszący	334

ROZDZIAŁ 23:	Strumień wejścia-wyjścia	337
	Jak działa strumień wejścia-wyjścia?	338
	Domyślne obiekty strumienia	338
	Strumień wejścia-wyjścia	340
	Tryby otwierania	341
	Hej, pliku, w jakim jesteś stanie?	342
	Czy możesz mi pokazać przykład?	343
	Inne metody klas obsługi strumieni	346
	Bezpośrednie czytanie i zapisywanie strumieni	348
	Zarządzanie formatem	350
	O co chodzi z endl?	351
	Umieszczanie wskaźnika w pliku	352
	Korzystanie z podklas klasy stringstream	353
	Manipulowanie manipulatorami	356

ROZDZIAŁ 24:	Obsługa błędów — wyjątki	359
	Uzasadnienie dla nowego mechanizmu obsługi błędów	361
	Działanie mechanizmu wyjątków	362
	Czym można rzucać?	365
	Uwaga, przechodzę!	367

ROZDZIAŁ 25:	Dziedziczenie wielokrotne	369
	Mechanizm dziedziczenia wielokrotnego	370
	Jak sobie poradzić z niejasnościami dotyczącymi dziedziczenia?	371
	Dziedziczenie wirtualne	372
	Konstruowanie obiektów dziedziczenia wielokrotnego	378
	Wyrażanie sprzecznych opinii	379
ROZDZIAŁ 26:	Szablony w C++	381
	Uogólnianie funkcji do szablonu	382
	Szablony klas	384
	Wskazówki dotyczące korzystania z szablonów	387
	Instancje szablonów zewnętrznych	388
	Implementowanie listy inicjalizacyjnej	388
ROZDZIAŁ 27:	Standaryzacja w bibliotece STL	391
	Kontener typu string	392
	Iterowanie po listach	397
	Przeglądanie listy	398
	Operacje na całej liście	400
	Czy możesz mi pokazać przykład?	400
ROZDZIAŁ 28:	Pisanie bezpiecznego kodu	403
	Motywy działania hakerów	404
	Na czym polega wstrzyknięcie kodu?	406
	Przykładowe wstrzyknięcie SQL	406
	Unikanie wstrzykiwania kodu	408
	Przepełnianie bufora dla zabawy i zysku	409
	Czy mogę zobaczyć przykład?	410
	Hakerskie wykorzystanie przepełnienia bufora	412
	Unikanie przepełnienia bufora — pierwsza próba	416
	Unikanie przepełnienia bufora — druga próba	418
	Kolejna zaleta klasy string	420
	Dlaczego nie zawsze należy używać klasy string?	422
CZĘŚĆ VI: DEKALOGI		425
ROZDZIAŁ 29:	Dziesięć sposobów na zapobieganie błędom	427
	Włączaj wszystkie ostrzeżenia i komunikaty o błędach	427
	Koduj w jasnym i spójnym stylu	428
	Ograniczaj widoczność	429
	Dodawaj komentarze do kodu w trakcie pisania	430

Wykonuj krok po kroku każdą ścieżkę przynajmniej raz	431
Unikaj przeciążania operatorów	431
Zarządzaj stertą systematycznie	431
Używaj wyjątków do obsługi błędów	432
Deklaruj wirtualne destrukторы	432
Unikaj dziedziczenia wielokrotnego	434

ROZDZIAŁ 30: Dziesięć sposobów na ochronę programów

przed hakerami435

Nie zakładaj niczego w odniesieniu	
do danych wprowadzanych przez użytkowników	436
Obsługuj awarie z wdziękiem	437
Zapisuj logi programu	438
Postępuj zgodnie z dobrymi zasadami wytwarzania oprogramowania	439
Wdrożenie kontroli wersji	440
Bezpiecznie uwierzytelniaj użytkowników	441
Zarządzaj zdalnymi sesjami	444
Zaciemniaj swój kod	445
Podpisz swój kod za pomocą certyfikatu cyfrowego	448
Użyj bezpiecznego szyfrowania wszędzie tam, gdzie jest to konieczne	449

CZĘŚĆ VII: DODATKI451

Skorowidz453