



Inne instrukcje C++ .....	62
Użycie obiektu cin .....	63
Złączanie za pomocą cout .....	64
cin i cout — klasy po raz pierwszy .....	64
Funkcje .....	66
Użycie funkcji zwracającej wartość .....	66
Odmiany funkcji .....	69
Funkcje definiowane przez użytkownika .....	70
Funkcje użytkownika zwracające wartość .....	73
Dyrektywa using w programach z wieloma funkcjami .....	74
Podsumowanie .....	76
Pytania sprawdzające .....	77
Ćwiczenia programistyczne .....	77
<b>Rozdział 3. Dane .....</b>	<b>79</b>
Zmienne proste .....	80
Nazwy zmiennych .....	80
Typy całkowitoliczbowe .....	81
Typy short, int, long i long long .....	82
Typy bez znaku .....	87
Dobór właściwego typu .....	89
Literały całkowitoliczbowe .....	90
Jak C++ ustala typ stałej? .....	91
Typ char — znaki i małe liczby całkowite .....	92
Typ danych bool .....	100
Kwalifikator const .....	100
Liczby zmiennoprzecinkowe .....	101
Zapis liczb zmiennoprzecinkowych .....	102
Zmiennoprzecinkowe typy danych .....	103
Stałe zmiennoprzecinkowe .....	105
Zalety i wady liczb zmiennoprzecinkowych .....	105
Operatory arytmetyczne C++ .....	106
Kolejność działań — priorytety operatorów i łączność .....	107
Odmiany dzielenia .....	108
Operator modulo .....	110
Konwersje typów .....	110
Automatyczne deklaracje typów w C++11 .....	116
Podsumowanie .....	117
Pytania sprawdzające .....	117
Ćwiczenia programistyczne .....	118
<b>Rozdział 4. Typy złożone .....</b>	<b>121</b>
Tablice w skrócie .....	122
Uwagi o programie .....	124
Inicjalizacja tablic .....	124
Inicjalizacja tablic w C++11 .....	125
Łańcuchy .....	126
Łączenie literałów napisowych .....	127
Łańcuchy w tablicy .....	128
Problemy z wprowadzaniem łańcuchów znakowych .....	129

800	Wczytywanie łańcuchów znakowych wierszami .....	130
804	Mieszanie w danych wejściowych łańcuchów i liczb .....	134
805	Klasa string — wprowadzenie .....	135
805	Inicjalizacja łańcuchów znakowych w C++11 .....	136
805	Przypisanie, konkatenacja i dołączanie .....	136
805	Inne operacje klasy string .....	138
805	Klasa string a wejście i wyjście .....	139
811	Inne odmiany literałów napisowych .....	141
811	Struktury .....	142
811	Użycie struktury w programie .....	143
811	Inicjalizacja struktur w C++11 .....	145
811	Czy w strukturze można użyć pola typu string? .....	146
811	Inne cechy struktur .....	146
811	Tablice struktur .....	148
811	Pola bitowe .....	149
811	Unie .....	149
811	Typy wyliczeniowe .....	151
811	Ustawianie wartości enumeratorów .....	153
811	Zakresy wartości w typach wyliczeniowych .....	153
811	Wskaźniki i różne drobiazgi .....	154
811	Deklarowanie i inicjalizacja wskaźników .....	156
811	Niebezpieczeństwa związane ze wskaźnikami .....	158
811	Wskaźniki i liczby .....	159
811	Użycie operatora new do alokowania pamięci .....	159
811	Zwalnianie pamięci za pomocą delete .....	161
811	Użycie new do tworzenia tablic dynamicznych .....	162
811	Wskaźniki, tablice i arytmetyka wskaźników .....	165
811	Podsumowanie informacji o wskaźnikach .....	168
811	Wskaźniki i łańcuchy .....	170
811	Użycie new do tworzenia struktur dynamicznych .....	174
811	Alokacja pamięci: automatyczna, statyczna i dynamiczna .....	177
811	Kombinacje typów .....	179
811	Tablice inaczej .....	181
811	Klasa szablonowa vector .....	181
811	Klasa szablonowa array (C++11) .....	182
811	Porównanie tablic z obiektami vector i array .....	183
811	Podsumowanie .....	184
811	Pytania sprawdzające .....	185
811	Ćwiczenia programistyczne .....	186
<b>811</b>	<b>Rozdział 5. Pętle i wyrażenia relacyjne .....</b>	<b>189</b>
811	Pętla for .....	190
811	Elementy pętli for .....	191
811	Wracamy do pętli for .....	196
811	Zmiana wielkości kroku .....	198
811	Pętla for i łańcuchy znakowe .....	198
811	Operatory inkrementacji (++) i dekrementacji (–) .....	199
811	Efekty uboczne i punkty odniesienia .....	200
811	Formy przedrostkowe a formy przyrostkowe .....	201

06C1	Operatory inkrementacji i dekrementacji a wskaźniki	202
06E1	Złożone operatory przypisania	203
06E1	Instrukcje złożone, czyli bloki	203
06E1	Przecinek jako operator (i pewne sztuczki składniowe)	205
06E1	Wyrażenia relacyjne	208
06E1	Przypisania, porównania i pomyłki	208
06E1	Porównywanie tańców w stylu C	210
10A1	Porównywanie obiektów klasy string	213
06E1	Pętla while	213
06A1	Uwagi o programie	215
06A1	Pętla for a pętla while	216
06A1	Chwilczkę — tworzymy pętlę opóźnienia	217
06A1	Pętla do while	219
06A1	Zakresowe pętle for (C++11)	221
06A1	Pętle i wprowadzanie danych tekstowych	221
06A1	Najprostszą wersją cin	222
10E1	cin.get(char) na odsiecz	223
02E1	Która wersja cin.get() jest lepsza?	224
02E1	Koniec pliku	224
06E1	Jeszcze inna wersja cin.get()	227
06E1	Pętle zagnieżdżone i dwuwymiarowe tablice	230
06E1	Inicjalizacja tablic dwuwymiarowych	232
06E1	Stosowanie tablic dwuwymiarowych	232
06E1	Podsumowanie	234
10E1	Pytania sprawdzające	234
02E1	Ćwiczenia programistyczne	235
	<b>Rozdział 6. Instrukcje warunkowe i operatory logiczne</b>	<b>237</b>
07E1	Instrukcja if	237
07E1	Instrukcja if else	239
07E1	Formatowanie instrukcji if else	241
07E1	Konstrukcja if else if else	241
10E1	Wyrażenia logiczne	243
10E1	Logiczny operator alternatywy —	243
10E1	Logiczny operator koniunkcji — &&	245
10E1	Ustalanie zakresu za pomocą operatora &&	247
10E1	Operator negacji logicznej — !	248
10E1	O operatorach logicznych	250
10E1	Zapis alternatywny	251
10E1	Biblioteka ctype	251
10E1	Operator ?:	253
10E1	Instrukcja switch	255
10E1	Użycie enumeratorów jako etykiet	258
10E1	switch versus if else	259
10E1	Instrukcje break i continue	259
10E1	Uwagi o programie	261
10E1	Pętla wczytywania liczb	262
10E1	Uwagi o programie	264
10E1	Formy przedstawiania i zapisu stałych	267

682	Proste wejście-wyjście z pliku .....	265
682	Tekstowe wejście-wyjście i pliki tekstowe .....	265
682	Zapis do pliku tekstowego .....	267
685	Odczyt danych z pliku tekstowego .....	270
686	Podsumowanie .....	274
686	Pytania sprawdzające .....	275
686	Ćwiczenia programistyczne .....	276
	<b>Rozdział 7. Funkcje — składniki programów w C++ .....</b>	<b>279</b>
676	Funkcje w skrócie .....	280
676	Definiowanie funkcji .....	281
676	Prototypowanie i wywoływanie funkcji .....	283
676	Parametry funkcji i przekazywanie przez wartość .....	286
682	Wiele parametrów .....	287
682	Jeszcze jedna funkcja dwuargumentowa .....	289
682	Funkcje i tablice .....	291
682	Jak wskaźniki umożliwiają tworzenie funkcji przetwarzających tablice? .....	292
682	Skutki użycia tablic jako parametrów .....	293
682	Dodatkowe przykłady funkcji i tablic .....	295
682	Funkcje korzystające z zakresów tablic .....	301
682	Wskaźniki i modyfikator const .....	302
682	Funkcje i tablice dwuwymiarowe .....	306
682	Funkcje i łańcuchy w stylu C .....	307
682	Funkcje z łańcuchami w stylu C jako parametrami .....	307
682	Funkcje zwracające łańcuchy w formacie C .....	309
682	Funkcje i struktury .....	310
682	Przekazywanie i zwracanie struktur .....	311
682	Inny przykład użycia funkcji i struktur .....	312
682	Przekazywanie adresu struktury .....	317
682	Funkcje i obiekty klasy string .....	318
682	Funkcje i obiekty typu array .....	320
682	Uwagi o programie .....	321
682	Rekurencja .....	322
682	Rekurencja w pojedynczym wywołaniu .....	322
682	Rekurencja w wielu wywołaniach .....	324
682	Wskaźniki na funkcje .....	325
682	Wskaźniki na funkcje — podstawy .....	325
682	Przykład użycia wskaźników na funkcje .....	327
682	Wariacje na temat wskaźników funkcji .....	329
682	Uproszczenie poprzez typedef .....	333
682	Podsumowanie .....	333
682	Pytania sprawdzające .....	334
682	Ćwiczenia programistyczne .....	336
	<b>Rozdział 8. Funkcje — zagadnienia zaawansowane .....</b>	<b>339</b>
682	Funkcje inline .....	339
682	Zmienne referencyjne .....	342
682	Tworzenie zmiennej referencyjnej .....	342
682	Referencje jako parametry funkcji .....	345
682	Właściwości referencji .....	348

085	Użycie referencji do struktur .....	352
086	Użycie referencji z obiektami .....	358
100	Obiekty po raz wtóry — obiekty, dziedziczenie i referencje .....	361
010	Kiedy korzystać z referencji jako parametrów? .....	364
012	Parametry domyślne .....	365
013	Uwagi o programie .....	366
014	Przeciążanie funkcji .....	367
	Przykład przeciążania funkcji .....	370
015	Kiedy korzystać z przeciążania funkcji? .....	372
081	Szablony funkcji .....	372
082	Przeciążone szablony .....	375
083	Ograniczenia szablonów .....	377
084	Specjalizacje jawne .....	377
085	Konkretyzacje i specjalizacje .....	380
086	Którą wersję funkcji wybierze kompilator? .....	382
087	Ewolucja szablonów funkcji .....	388
088	Podsumowanie .....	392
089	Pytania sprawdzające .....	392
090	Ćwiczenia programistyczne .....	393
	<b>Rozdział 9. Model pamięci i przestrzenie nazw .....</b>	<b>397</b>
000	Kompilacja rozłączna .....	397
100	Czas życia, zasięg i łączenie .....	403
101	Zasięg i łączenie .....	404
010	Przydział automatyczny .....	404
011	Zmienne statyczne .....	409
012	Przydział statyczny, łączenie zewnętrzne .....	411
013	Specyfikatory i kwalifikatory .....	419
014	Łączenie a funkcje .....	421
015	Łączenie językowe .....	422
016	Kategorie przydziału a przydział dynamiczny .....	423
017	Przestrzenie nazw .....	429
018	Tradycyjne przestrzenie nazw języka C++ .....	429
019	Nowe mechanizmy przestrzeni nazw .....	431
020	Przestrzenie nazw — przykład .....	438
021	Przyszłość przestrzeni nazw .....	441
022	Podsumowanie .....	442
023	Pytania sprawdzające .....	442
024	Ćwiczenia programistyczne .....	445
	<b>Rozdział 10. Obiekty i klasy .....</b>	<b>447</b>
	Programowanie proceduralne a programowanie obiektowe .....	448
000	Klasy a abstrakcje .....	449
001	Czym jest typ? .....	449
002	Klasy w języku C++ .....	450
003	Implementowanie metod klas .....	455
004	Stosowanie klas .....	459
005	Zmiany implementacji .....	461
006	Podsumowanie poznanych wiadomości .....	462

463	Konstruktory i destruktory .....	463
464	Deklarowanie i definiowanie konstruktorów .....	464
465	Stosowanie konstruktorów .....	465
466	Konstruktory domyślne .....	466
467	Destruktry .....	467
468	Ulepszenia klasy Stock .....	468
475	Konstruktory i destruktory — podsumowanie .....	475
476	Tożsamość obiektu — wskaźnik this .....	476
482	Tablice obiektów .....	482
485	Zasięg klasy .....	485
486	Stałe zasięgu klasy .....	486
487	Wyliczenia z własnym zasięgiem (C++11) .....	487
488	Abstrakcyjne typy danych .....	488
492	Podsumowanie .....	492
493	Pytania sprawdzające .....	493
493	Ćwiczenia programistyczne .....	493
<b>497</b>	<b>Rozdział 11. Stosowanie klas .....</b>	<b>497</b>
498	Przeciążanie operatorów .....	498
499	Raz, dwa, trzy — próba przeciążenia operatora .....	499
502	Dodatkowy operator dodawania .....	502
505	Ograniczenia przeciążania operatorów .....	505
506	Jeszcze o przeciążaniu operatorów .....	506
509	Przyjaciele najważniejsi .....	509
510	Deklarowanie przyjaźni .....	510
512	Typowa przyjaźń — przeciążanie operatora << .....	512
518	Przeciążanie operatorów — metody kontra funkcje nieskładowe .....	518
519	Przeciążania ciąg dalszy — klasa Vector .....	519
526	Składowa kodująca stan obiektu .....	526
528	Przeciążanie operatorów arytmetycznych dla klasy Vector .....	528
530	Nota implementacyjna .....	530
530	Wektorowe błędnie losowe .....	530
534	Automatyczne konwersje i rzutowanie typów klas .....	534
539	O programie .....	539
539	Funkcje konwersji .....	539
544	Konwersja a zaprzyjaźnienie .....	544
547	Podsumowanie .....	547
549	Pytania sprawdzające .....	549
549	Ćwiczenia programistyczne .....	549
<b>553</b>	<b>Rozdział 12. Klasy a dynamiczny przydział pamięci .....</b>	<b>553</b>
554	Klasy a pamięć dynamiczna .....	554
554	Powtórka z pamięci dynamicznej i statyczne składowe klas .....	554
562	Specjalne metody klasy .....	562
565	W czym tkwi problem z konstruktorem kopiującym w Stringbad? .....	565
568	Kolejne słabości Stringbad: operatory przypisania .....	568
571	Nowa, ulepszona klasa — String .....	571
572	Nowa wersja konstruktora domyślnego .....	572
573	Porównywanie ciągów .....	573

584	Indeksowanie ciągu .....	574
184	Statische metody klasy .....	575
584	Dalsze przeciążanie operatora przypisania .....	576
804	O czym należy pamiętać, stosując new w konstruktorach? .....	581
184	Zalecenia i przestrogi .....	582
804	Kopiowanie obiektów składowa po składowej .....	583
274	Słów parę o zwracaniu obiektów .....	584
874	Zwracanie niemodyfikowalnej (const) referencji obiektu .....	584
584	Zwracanie modyfikowalnej (bez const) referencji do obiektu .....	585
284	Zwracanie obiektu przez wartość .....	585
884	Zwracanie przez wartość obiektu niemodyfikowalnego (const) .....	586
184	Wskaźniki obiektów .....	587
844	Jeszcze o new i delete .....	589
524	Wskaźniki obiektów — podsumowanie .....	590
884	Jeszcze o miejscowej wersji new .....	592
624	Powtórka z poznanych technik .....	596
784	Przeciążanie operatora << .....	596
844	Funkcje konwersji .....	597
184	Klasy wykorzystujące new w konstruktorach .....	597
524	Symulacja kolejki .....	598
824	Klasa kolejki .....	598
804	Klasa klienta .....	609
804	Symulacja bankomatu .....	612
804	Podsumowanie .....	616
812	Pytania sprawdzające .....	617
812	Ćwiczenia programistyczne .....	619
	<b>Rozdział 13. Klasy i dziedziczenie .....</b>	<b>623</b>
824	Prosta klasa bazowa .....	624
824	Dziedziczenie .....	626
824	Konstruktory — zagadnienia związane z poziomem dostępu .....	628
824	Korzystanie z klasy pochodnej .....	631
824	Relacje między klasą pochodną a bazową .....	633
824	Dziedziczenie — relacja jest-czymś .....	635
824	Polimorficzne dziedziczenie publiczne .....	636
844	Tworzenie klas Brass oraz BrassPlus .....	637
844	Wiązanie statyczne i dynamiczne .....	648
844	Zgodność typów wskaźnikowych i referencyjnych .....	648
844	Metody wirtualne i wiązanie dynamiczne .....	650
844	Co trzeba wiedzieć o metodach wirtualnych? .....	653
844	Kontrola dostępu — poziom chroniony .....	656
844	Abstrakcyjne klasy bazowe .....	657
844	Stosowanie abstrakcyjnych klas bazowych .....	659
844	Filozofia abstrakcyjnych klas bazowych .....	665
844	Dziedziczenie i dynamiczny przydział pamięci .....	665
844	Przypadek pierwszy — klasa pochodna bez dynamicznego przydziału pamięci .....	665
844	Przypadek drugi — klasa pochodna z dynamicznym przydziałem pamięci .....	666
844	Przykład dziedziczenia z wykorzystaniem dynamicznego przydziału pamięci oraz funkcji zaprzyjaźnionych .....	668

687	Projektowanie klas — przegląd zagadnień .....	673
687	Metody automatycznie generowane przez kompilator .....	673
687	Inne metody .....	675
688	Dziedziczenie publiczne .....	678
688	Metody klasy — podsumowanie .....	682
688	Podsumowanie .....	683
688	Pytania sprawdzające .....	683
688	Ćwiczenia programistyczne .....	684
<b>687</b>	<b>Rozdział 14. Wielokrotne użycie kodu w C++ .....</b>	<b>687</b>
688	Klasy ze składowymi w postaci obiektów .....	688
688	Krótka charakterystyka klasy valarray .....	688
689	Projekt klasy Student .....	689
691	Przykładowa klasa Student .....	691
697	Dziedziczenie prywatne .....	697
697	Nowa wersja klasy Student .....	697
706	Dziedziczenie wielokrotne .....	706
711	Podwójne egzemplarze klasy Worker .....	711
714	Podwójne metody .....	714
723	Przegląd zagadnień związanych z dziedziczeniem wielokrotnym .....	723
724	Szablony klas .....	724
724	Definiowanie szablonu klasy .....	724
727	Korzystanie z szablonu klasy .....	727
729	Analiza szablonu klasy .....	729
734	Szablon tablicy i argumenty pozatypowe szablonu .....	734
736	Elastyczność szablonów .....	736
739	Specjalizacja szablonu .....	739
742	Szablony jako składowe .....	742
744	Szablony jako parametry .....	744
746	Szablony klas i zaprzyjaźnienie .....	746
752	Szablonowe aliasy typów (C++11) .....	752
753	Podsumowanie .....	753
755	Pytania sprawdzające .....	755
757	Ćwiczenia programistyczne .....	757
<b>763</b>	<b>Rozdział 15. Zaprzyjaźnienie, wyjątki i nie tylko .....</b>	<b>763</b>
763	Zaprzyjaźnienie .....	763
764	Klasy zaprzyjaźnione .....	764
768	Zaprzyjaźnione metody klas .....	768
771	Inne relacje przyjaźni .....	771
773	Klasy zagnieżdżone .....	773
774	Dostęp do klas zagnieżdżonych .....	774
776	Zagnieżdżanie w szablonie .....	776
779	Wyjątki .....	779
779	Wywoływanie funkcji abort() .....	779
780	Zwracanie kodu błędu .....	780
782	Mechanizm wyjątków .....	782
784	Wyjątki w postaci obiektów .....	784
788	Specyfikacje wyjątków a C++11 .....	788
789	Rozwijanie stosu .....	789

870	Inne właściwości wyjątków	793
870	Klasa exception	796
870	Wyjątki, klasy i dziedziczenie	799
870	Problemy z wyjątkami	804
880	Ostrożnie z wyjątkami	807
880	RTTI	808
880	Po co nam RTTI?	808
880	Jak działa RTTI?	809
880	Operatory rzutowania typu	816
880	Podsumowanie	820
880	Pytania sprawdzające	820
880	Ćwiczenia programistyczne	822
	<b>Rozdział 16. Klasa string oraz biblioteka STL</b>	<b>823</b>
880	Klasa string	823
880	Tworzenie obiektu string	824
880	Wprowadzanie danych do obiektów string	828
880	Używanie obiektów string	830
880	Co jeszcze oferuje klasa string?	835
880	Warianty klasy string	837
880	Szablony klas inteligentnych wskaźników	837
880	Stosowanie inteligentnych wskaźników	838
880	Więcej o inteligentnych wskaźnikach	841
880	Wyższość unique_ptr nad auto_ptr	844
880	Wybór inteligentnego wskaźnika	845
880	Biblioteka STL	847
880	Szablon klasy vector	847
880	Metody klasy vector	849
880	Inne możliwości klasy vector	853
880	Zakresowe pętle for (C++11)	857
880	Programowanie uogólnione	858
880	Do czego potrzebne są iteratory?	858
880	Rodzaje iteratorów	862
880	Hierarchia iteratorów	865
880	Pojęcia, uściślenia i modele	866
880	Rodzaje kontenerów	872
880	Kontenery asocjacyjne	881
880	Nieuporządkowane kontenery asocjacyjne (C++11)	887
880	Obiekty funkcyjne (funktory)	887
880	Pojęcia związane z funktorami	888
880	Funktory predefiniowane	891
880	Funktory adaptowalne i adaptatory funkcji	892
880	Algorytmy	895
880	Grupy algorytmów	895
880	Ogólne właściwości algorytmów	896
880	Biblioteka STL i klasa string	897
880	Funkcje a metody kontenerów	898
880	Używanie biblioteki STL	899

Inne biblioteki .....	903
Klasy vector, valarray i array .....	903
Szablon initializer_list (C++11) .....	908
Stosowanie szablonu initializer_list .....	910
Uwagi do programu .....	911
Podsumowanie .....	911
Pytania sprawdzające .....	913
Ćwiczenia programistyczne .....	914
<b>Rozdział 17. Obsługa wejścia, wyjścia oraz plików .....</b>	<b>917</b>
Ogólna charakterystyka obsługi wejścia-wyjścia w języku C++ .....	918
Strumienie i bufory .....	919
Strumienie i bufory a plik istream .....	921
Przekierowanie .....	923
Realizacja operacji wyjścia z wykorzystaniem obiektu cout .....	924
Przeciążony operator << .....	924
Inne metody klasy ostream .....	927
Opróżnianie bufora wyjściowego .....	930
Formatowanie danych wyjściowych za pomocą obiektu cout .....	931
Realizacja operacji wejścia z wykorzystaniem obiektu cin .....	945
Jak operator >> obiektu cin „widzi” dane wejściowe? .....	947
Stany strumienia .....	949
Inne metody klasy istream .....	953
Pozostałe metody klasy istream .....	960
Wejście-wyjście plikowe .....	964
Proste operacje wejścia-wyjścia plikowego .....	965
Kontrola strumienia i metoda is_open() .....	968
Otwieranie wielu plików .....	969
Przetwarzanie argumentów wiersza polecenia .....	969
Tryby otwarcia pliku .....	971
Dostęp swobodny .....	981
Formatowanie wewnętrzne .....	988
Podsumowanie .....	991
Pytania sprawdzające .....	992
Ćwiczenia programistyczne .....	993
<b>Rozdział 18. Nowy standard C++ .....</b>	<b>997</b>
Podsumowanie omawianych elementów C++11 .....	997
Nowe typy .....	997
Jednolita inicjalizacja .....	998
Deklaracje .....	999
nullptr .....	1001
Inteligentne wskaźniki .....	1002
Zmiany w specyfikacji wyjątków .....	1002
Jawny zasięg elementów wliczeń .....	1002
Zmiany w klasach .....	1003
Zmiany w szablonach i bibliotece STL .....	1004
Referencje r-wartościowe .....	1006

101  
104  
10609

808	Semantyka przeniesienia i referencje r-wartościowe	1007
809	Potrzeba semantyki przeniesienia	1007
809	Przykład przenoszenia	1008
810	Konstruktor przenoszący — wnioski	1013
810	Przypisania	1014
810	Wymuszanie przeniesienia	1015
810	Nowe elementy klas	1018
810	Specjalne metody klas	1018
810	Metody domyślne i usunięte	1019
810	Delegowanie konstruktorów	1021
810	Dziedziczenie konstruktorów	1021
810	Zarządzanie metodami wirtualnymi: override i final	1023
810	Funkcje lambda	1024
810	Wskaźniki do funkcji, funktory i lambdy	1024
810	Po co nam lambdy?	1027
810	Adaptory	1030
810	Adapter funkcji a nieefektywność szablonów	1030
810	Naprawa problemu	1032
810	Dalsze możliwości	1034
810	Szablony o zmiennej liczbie parametrów	1035
810	Pakiety parametrów szablonu i funkcji	1035
810	Rozpakowywanie pakietów	1036
810	Rekurencja w szablonach o zmiennej liczbie parametrów	1037
810	Pozostałe udogodnienia C++11	1040
810	Programowanie współbieżne	1040
810	Uzupełnienia biblioteki	1040
810	Programowanie niskopoziomowe	1041
810	Inne	1042
810	Zmiany języka	1042
810	Projekt Boost	1043
810	TR1	1043
810	Korzystanie z bibliotek Boost	1043
810	Co dalej?	1044
810	Podsumowanie	1045
810	Pytania sprawdzające	1046
810	Ćwiczenia programistyczne	1049
810	<b>Dodatek A Systemy liczbowe</b>	<b>1051</b>
810	Liczby dziesiętne (o podstawie 10)	1051
810	Liczby całkowite ósemkowe (o podstawie 8)	1051
810	Liczby szesnastkowe	1052
810	Liczby dwójkowe (o podstawie 2)	1052
810	Zapis dwójkowy a szesnastkowy	1053
810	<b>Dodatek B Słowa zastrzeżone języka C++</b>	<b>1055</b>
810	Słowa kluczowe języka C++	1055
810	Leksyemy alternatywne	1056
810	Nazwy zastrzeżone bibliotek języka C++	1056
810	Identyfikatory o specjalnym znaczeniu	1057

<b>Dodatek C</b>	<b>Zestaw znaków ASCII</b> .....	<b>1059</b>
<b>Dodatek D</b>	<b>Priorytety operatorów</b> .....	<b>1063</b>
<b>Dodatek E</b>	<b>Inne operatory</b> .....	<b>1067</b>
	Operatory bitowe .....	1067
	Operatory przesunięcia .....	1067
	Bitowe operatory logiczne .....	1069
	Alternatywne reprezentacje operatorów bitowych .....	1071
	Kilka typowych technik wykorzystujących operatory bitowe .....	1072
	Operatory wyłuskania składowych .....	1073
	alignof (C++11) .....	1077
	noexcept (C++11) .....	1078
<b>Dodatek F</b>	<b>Klasa szablonowa string</b> .....	<b>1079</b>
	Trzyście typów i stała .....	1080
	Informacje o danych, konstruktory i różne drobiazgi .....	1080
	Konstruktor domyślny .....	1083
	Konstruktory operujące na klasycznych łańcuchach C .....	1083
	Konstruktory operujące na fragmentach łańcuchów C .....	1084
	Konstruktory operujące na referencji l-wartościowej .....	1084
	Konstruktory operujące na referencji r-wartościowej (C++11) .....	1085
	Konstruktory wykorzystujące n kopii znaku .....	1086
	Konstruktory wykorzystujące zakres .....	1086
	Konstruktor operujący na liście inicjalizującej (C++11) .....	1086
	Metody zarządzające pamięcią .....	1087
	Dostęp do łańcucha .....	1087
	Proste przypisanie .....	1088
	Przeszukiwanie łańcuchów .....	1089
	Rodzina funkcji find() .....	1089
	Rodzina funkcji rfind() .....	1089
	Rodzina funkcji find_first_of() .....	1090
	Rodzina funkcji find_last_of() .....	1090
	Rodzina funkcji find_first_not_of() .....	1091
	Rodzina funkcji find_last_not_of() .....	1091
	Metody i funkcje porównania .....	1091
	Modyfikatory łańcuchów .....	1093
	Metody dołączania i dodawania .....	1093
	Inne metody przypisania .....	1094
	Metody wstawiania .....	1094
	Metody usuwania .....	1095
	Metody zastępowania .....	1095
	Pozostałe metody modyfikujące: copy() oraz swap() .....	1096
	Wejście i wyjście .....	1096
<b>Dodatek G</b>	<b>Metody i funkcje z biblioteki STL</b> .....	<b>1099</b>
	STL a C++11 .....	1099
	Nowe kontenery .....	1099
	Zmiany w kontenerach C++98 .....	1100
	Składowe wspólne dla wszystkich (lub większości) kontenerów .....	1101
	Dodatkowe składowe dla kontenerów sek	1104

1080	Dodatkowe operacje zbiorów i map .....	1107
1081	Kontenery asocjacyjne nieporządkujące (C++11) .....	1109
1082	Funkcje STL .....	1111
1083	Niemodyfikujące operacje sekwencyjne .....	1111
1084	Mutujące operacje sekwencyjne .....	1116
1085	Operacje sortowania i pokrewne .....	1125
1086	Operacje liczbowe .....	1139
	<b>Dodatek H Wybrane pozycje książkowe i zasoby internetowe .....</b>	<b>1141</b>
1087	Wybrane pozycje książkowe .....	1141
1088	Zasoby internetowe .....	1142
	<b>Dodatek I Dostosowywanie do standardu ANSI/ISO C++ .....</b>	<b>1145</b>
1089	Unikanie nadużywania niektórych dyrektyw preprocesora .....	1145
1090	Do definiowania stałych lepiej używać modyfikatora const	
1091	niż dyrektywy #define .....	1145
1092	Do definiowania niewielkich funkcji lepiej używać specyfikatora inline niż	
1093	makrodefinicji #define .....	1147
1094	Używanie prototypów funkcji .....	1148
1095	Stosowanie rzutowania typów .....	1148
1096	Poznanie i wykorzystywanie mechanizmów języka C++ .....	1149
1097	Używanie nowej organizacji plików nagłówkowych .....	1149
1098	Korzystanie z przestrzeni nazw .....	1149
1099	Używanie inteligentnych wskaźników .....	1150
1100	Używanie klasy string .....	1151
1101	Korzystanie z biblioteki STL .....	1151
	<b>Dodatek J Odpowiedzi do pytań sprawdzających .....</b>	<b>1153</b>
	<b>Skorowidz .....</b>	<b>1179</b>