

O autorze	XV
O redaktorze technicznym	XVII
Podziękowania	XVII
Wprowadzenie	XVIII
Tworzymy zespół	XVIII
Przegląd zawartości książki	XIX
Kod źródłowy do książki	XXVI
Aktualizacje	XXVII
Kontakt ze mną	XXVII
CZĘŚĆ 1. Wprowadzenie do języka C# i platformy .NET	1
ROZDZIAŁ 1. Filozofia platformy .NET	3
Omówienie dotychczasowego stanu rzeczy	3
Rozwiązanie .NET	6
Wprowadzenie do bloków budulcowych platformy .NET: CLR, CTS i CLS	6
Inne języki programowania na platformie .NET	9
Przegląd pakietów .NET	11
Wspólny system typów (CTS)	16
Wspólna specyfikacja językowa (CLS)	19
Wspólne środowisko uruchomieniowe (CLR)	21
Różnica między pakietem, przestrzenią nazw i typem	22
Analiza pakietu w programie ildasm.exe	27
Analiza pakietu w programie Reflector	29
Instalowanie środowiska uruchomieniowego .NET	30
Niezależność .NET od platformy systemowej	30
Podsumowanie	32
ROZDZIAŁ 2. Tworzenie aplikacji w C#	33
Rola .NET Framework 4.0 SDK	33
Tworzenie aplikacji C# za pomocą csc.exe	34
Tworzenie aplikacji .NET w programie Notepad++	39
Tworzenie aplikacji .NET w SharpDevelop	40
Tworzenie aplikacji .NET w Visual C# 2010 Express	42
Tworzenie aplikacji .NET w Visual Studio 2010	43
Podsumowanie	54
CZĘŚĆ 2. Podstawowe konstrukcje programistyczne C#	55
ROZDZIAŁ 3. Podstawowe konstrukcje programistyczne C#. Część 1	57
Anatomia prostego programu w języku C#	57
Ciekawostka: kilka dodatkowych składowych klasy System.Environment	62
Klasa System.Console	64
Systemowe typy danych i skrótowa notacja C#	68
Używanie danych typu string	76
Zawężające i poszerzające konwersje typów danych	82

Niejawna typizacja zmiennych lokalnych	88
Konstrukcje iteracyjne w języku C#	92
Konstrukcje decyzyjne oraz operatory relacyjne	94
Podsumowanie	96
ROZDZIAŁ 4. Podstawowe konstrukcje programistyczne C#. Część 2	97
Metody i modyfikatory parametrów	97
Tablice w języku C#	107
Wyliczenia	113
Struktury	118
Typy wartościowe i typy referencyjne	121
Typy nullable w języku C#	127
Podsumowanie	130
ROZDZIAŁ 5. Klasy i enkapsulacja	131
Wprowadzenie do klas w języku C#	131
Konstruktory	134
Rola słowa kluczowego this	137
Rola słowa kluczowego static	143
Trzy filary programowania obiektowego	149
Modyfikatory dostępu w języku C#	152
Pierwszy filar programowania obiektowego: enkapsulacja w języku C#	154
Właściwości automatyczne	165
Składnia inicjalizacji obiektów	168
Pola danych stałych	171
Typy częściowe	174
Podsumowanie	175
ROZDZIAŁ 6. Dziedziczenie i polimorfizm	177
Podstawowa mechanika dziedziczenia	177
Powtórka z diagramów klas w Visual Studio	181
Drugi filar programowania obiektowego: szczegółowe informacje na temat dziedziczenia	182
Programowanie z wykorzystaniem modelu zawierania/delegacji	187
Trzeci filar programowania obiektowego: polimorfizm w języku C#	190
Reguły rzutowania między klasą bazową a klasą potomną	201
Najważniejsza klasa nadrzędna: System.Object	204
Podsumowanie	210
ROZDZIAŁ 7. Strukturalna obsługa wyjątków	211
Oda do błędów, bugów i wyjątków	211
Rola obsługi wyjątków na platformie .NET	212
Przykład najprostszy z możliwych	214
Konfigurowanie stanu wyjątku	218
Wyjątki systemowe (System.SystemException)	222

Wyjątki na poziomie aplikacji (System.ApplicationException)	223
Przetwarzanie wielu wyjątków	226
Kto zgłasza co?	231
Konsekwencje nieobsłużenia wyjątków	232
Debugowanie nieobsłużonych wyjątków w Visual Studio	232
Słowo na temat wyjątków CSE	233
Podsumowanie	235
ROZDZIAŁ 8. Czas życia obiektu	237
Klasy, obiekty i referencje	237
Podstawowe informacje o czasie życia obiektu	238
Rola korzeni aplikacji	241
Generacje obiektów	242
Odśmiecanie współbieżne na platformie .NET 1.0–3.5	243
Typ System.GC	244
Tworzenie obiektów finalizowalnych	247
Tworzenie obiektów jednorazowych	250
Tworzenie obiektów finalizowalnych i jednorazowych	253
Leniwe tworzenie instancji obiektów	256
Podsumowanie	259
CZĘŚĆ 3. Zaawansowane konstrukcje programistyczne C#	261
ROZDZIAŁ 9. Interfejsy	263
Wprowadzenie do interfejsów	263
Definiowanie interfejsów niestandardowych	266
Implementowanie interfejsu	268
Wywoływanie składowych interfejsu na poziomie obiektu	270
Interfejsy jako parametry	272
Interfejsy jako wartości zwracane	273
Tablice interfejsów	274
Implementowanie interfejsów w Visual Studio 2010	275
Rozwiązywanie konfliktów nazw za pomocą jawnej implementacji interfejsów	276
Projektowanie hierarchii interfejsów	279
Tworzenie obiektów wyliczalnych (IEnumerable oraz IEnumerator)	282
Tworzenie obiektów klonowalnych (ICloneable)	287
Tworzenie obiektów porównywalnych (Comparable)	291
Podsumowanie	296
ROZDZIAŁ 10. Typy generyczne	297
Problemy z kolekcjami niegenerycznymi	297
Rola parametrów typów generycznych	306
Przestrzeń nazw System.Collections.Generic	309
Tworzenie niestandardowych metod generycznych	317
Tworzenie niestandardowych generycznych struktur i klas	320

Ograniczenia parametrów typu	323
Podsumowanie	325
ROZDZIAŁ 11. Delegaty, zdarzenia i lambdy	327
Delegaty .NET	327
Definiowanie delegatów w języku C#	328
Klasy bazowe System.MulticastDelegate oraz System.Delegate	330
Najprostszy przykład delegacji	331
Wysyłanie powiadomień o stanie obiektu za pomocą delegatów	334
Składnia grupowej konwersji metod	339
Kowariancja delegatów	340
Delegaty generyczne	342
Zdarzenia w języku C#	344
Metody anonimowe w języku C#	352
Wyrażenia lambda	355
Podsumowanie	361
ROZDZIAŁ 12. Zaawansowane elementy języka C#	363
Indeksatory	363
Przeciążanie operatorów	368
Konwersja typów niestandardowych	376
Metody rozszerzające	382
Metody częściowe	390
Typy anonimowe	392
Typy wskaźnikowe	397
Podsumowanie	404
ROZDZIAŁ 13. LINQ to Objects	405
Konstrukcje programistyczne typowe dla LINQ	405
Rola LINQ	408
Stosowanie zapytań LINQ do prostych tablic	410
Zwracanie wyniku zapytania LINQ	416
Stosowanie zapytań LINQ do kolekcji obiektów	418
Operatory zapytaniowe LINQ	421
Wewnętrzna reprezentacja instrukcji zapytań LINQ	429
Podsumowanie	433
CZEŚĆ 4. Programowanie z wykorzystaniem pakietów .NET	435
ROZDZIAŁ 14. Konfigurowanie pakietów .NET	437
Definiowanie niestandardowych przestrzeni nazw	437
Rola pakietów .NET	443
Format pakietu .NET	444
Tworzenie i użytkowanie pakietu jednoplikowego	448
Tworzenie i użytkowanie pakietu wieloplikowego	457
Pakiety prywatne	460

Pakiety współdzielone	464
Użytkowanie pakietów współdzielonych	470
Konfigurowanie pakietów współdzielonych	472
Pakiety z zasadami wydawcy	476
Element	477
Przestrzeń nazw System.Configuration	479
Podsumowanie	480
ROZDZIAŁ 15. Refleksja typów, późne wiązanie i programowanie z wykorzystaniem atrybutów	481
Niezbędność metadanych typu	481
Refleksja	486
Tworzenie niestandardowej przeglądarki metadanych	488
Dynamiczne ładowanie pakietów	493
Refleksja pakietów współdzielonych	496
Późne wiązanie	497
Rola atrybutów .NET	501
Tworzenie atrybutów niestandardowych	504
Atrybuty pakietów i modułów	507
Refleksja atrybutów przy wczesnym wiązaniu	508
Refleksja atrybutów przy późnym wiązaniu	509
Refleksja, późne wiązanie i atrybuty niestandardowe – podsumowanie	511
Tworzenie rozszerzalnej aplikacji	511
Podsumowanie	516
ROZDZIAŁ 16. Procesy, domeny aplikacji i konteksty obiektów	517
Rola procesu w systemie Windows	517
Komunikacja z procesami na platformie .NET	519
Domeny aplikacji .NET	527
Komunikacja z domyślną domeną aplikacji	529
Tworzenie nowych domen aplikacji	532
Granice kontekstowe obiektów	535
Podsumowanie procesów, domen aplikacji i kontekstów	539
Podsumowanie	539
ROZDZIAŁ 17. Język CIL i rola pakietów dynamicznych	541
Dlaczego warto poznać gramatykę CIL	541
Analiza dyrektyw, atrybutów i kodów operacyjnych	542
Wrzucanie i rzucanie: stosowy charakter języka CIL	544
Inżynieria wahadłowa	545
Dyrektywy i atrybuty CIL	552
Biblioteka klas bazowych .NET, język C# i mapowanie typów danych CIL	558
Definiowanie składowych typu w języku CIL	558
Definiowanie pól danych w języku CIL	558

Analiza kodów operacyjnych języka CIL	561
Tworzenie pakietu .NET w języku CIL	566
Pakiety dynamiczne	570
Podsumowanie	578
ROZDZIAŁ 18. Typy dynamiczne i środowisko DLR (Dynamic Language Runtime)	579
Rola słowa kluczowego dynamic w języku C#	579
Rola środowiska DLR (Dynamic Language Runtime)	584
Upraszczenie wywołań z późnym wiązaniem za pomocą typów dynamicznych	586
Łatwiejsza współpraca z COM dzięki danym dynamicznym	589
Współpraca z COM z wykorzystaniem elementów języka C# 4.0	592
Podsumowanie	597
CZĘŚĆ 5. Wprowadzenie do bibliotek klas bazowych .NET	599
ROZDZIAŁ 19. Programowanie wielowątkowe i równoległe	601
Zależności między procesem, domeną aplikacji, kontekstem i wątkiem	601
Krótką powtórka z delegatów .NET	603
Asynchroniczna natura delegatów	605
Asynchroniczne wywoływanie metod	606
Przestrzeń nazw System.Threading	611
Klasa System.Threading.Thread	612
Tworzenie wątków dodatkowych w kodzie programu	615
Kwestia współbieżności	620
Programowanie z wykorzystaniem delegatów TimerCallback	626
Pula wątków CLR	628
Programowanie równoległe na platformie .NET	629
Równoległe zapytania LINQ (PLINQ)	637
Podsumowanie	640
ROZDZIAŁ 20. Plikowe operacje wejścia/wyjścia oraz serializacja obiektów	641
Przestrzeń nazw System.IO	641
Typ DirectoryInfo oraz FileInfo	642
Typ DirectoryInfo	643
Typ Directory	646
Typ DriveInfo	647
Typ FileInfo	648
Typ File	652
Abstrakcyjna klasa Stream	654
Typy StreamWriter oraz StreamReader	656
Typy StringWriter oraz StringReader	658
Typy BinaryWriter oraz BinaryReader	660
Monitorowanie plików w kodzie programu	661
Serializacja obiektów	663

Przygotowanie obiektów do serializacji	665
Wybór formatera serializacji	667
Serializowanie obiektów za pomocą typu BinaryFormatter	669
Serializowanie obiektów za pomocą typu SoapFormatter	671
Serializowanie obiektów za pomocą typu XmlSerializer	672
Serializowanie kolekcji obiektów	674
Dostosowywanie serializacji Soap/binarnej	675
Podsumowanie	680
ROZDZIAŁ 21. ADO.NET. Część I. Warstwa połączeniowa	681
Ogólna definicja ADO.NET	681
Trzy oblicza ADO.NET	682
Dostawcy danych ADO.NET	683
Dodatkowe przestrzenie nazw ADO.NET	686
Typy z przestrzeni nazw System.Data	686
Ukrywanie dostawców danych za pomocą interfejsów	690
Tworzenie bazy danych AutoLot	693
Model fabryki dostawców danych ADO.NET	698
Warstwa połączeniowa ADO.NET	703
Używanie typów DataReader	708
Tworzenie biblioteki dostępu do danych wielokrotnego użytku	710
Tworzenie nakładki na aplikację konsolową	718
Omówienie transakcji bazodanowych	723
Podsumowanie	728
ROZDZIAŁ 22. ADO.NET. Część II. Warstwa bezpołączeniowa	729
Warstwa bezpołączeniowa ADO.NET	729
Rola typu DataSet	730
Typy DataRow	735
Typy DataTable	738
Dowiązywanie obiektów DataTable do graficznych interfejsów użytkownika Windows Forms	743
Używanie adapterów danych	751
Dodawanie funkcji bezpołączeniowych do AutoLotDAL.dll	754
Wielotabelowe obiekty DataSet i relacje danych	758
Narzędzia do projektowania baz danych Windows Forms	762
Wyodrębnianie kodu bazy danych ze ścisłą kontrolą typów do biblioteki klas	770
Programowanie z wykorzystaniem LINQ to DataSet	775
Podsumowanie	780
ROZDZIAŁ 23. ADO.NET. Część III. The Entity Framework	781
Rola Entity Framework	781
Tworzenie i analiza pierwszego EDM	789
Programowanie z wykorzystaniem modelu koncepcyjnego	798

AutoLotDAL wersja 4.0, teraz już z encjami	803
Encje dowiązujące dane do interfejsów graficznych Windows Forms	808
Podsumowanie	812
ROZDZIAŁ 24. Wprowadzenie do LINQ to XML	813
Opowieść o dwóch API XML	813
Składowe przestrzeni nazw System.Xml.Linq	816
Używanie typów XElement oraz XDocument	820
Operacje na dokumentach XML znajdujących się w pamięci	823
Podsumowanie	827
ROZDZIAŁ 25. Wprowadzenie do WCF (Windows Communication Foundation)	829
Potpourri rozproszonych API obliczeniowych	829
Rola WCF	835
Analiza najważniejszych pakietów WCF	837
Szablony projektów WCF w Visual Studio	838
Podstawowa budowa aplikacji WCF	840
ABC usług WCF	841
Tworzenie usługi WCF	845
Hostowanie usługi WCF	848
Tworzenie aplikacji klienckiej WCF	855
Upraszczenie ustawień konfiguracyjnych w WCF 4.0	859
Używanie szablonu projektu biblioteki usługi WCF	865
Hostowanie usługi WCF jako usługi Windows	868
Asynchroniczne wywoływanie usługi z poziomu klienta	872
Projektowanie kontraktów danych WCF	874
Podsumowanie	879
ROZDZIAŁ 26. Wprowadzenie do Windows Workflow Foundation 4.0	881
Definiowanie procesu biznesowego	881
Tworzenie bardzo prostego przepływu pracy	882
Środowisko uruchomieniowe WF 4.0	886
Omówienie aktywności Workflow 4.0	890
Tworzenie schematu blokowego przepływu pracy	893
Wyodrębnianie przepływów pracy do dedykowanych bibliotek	901
Użytkowanie biblioteki przepływu pracy	908
Podsumowanie	910
CZĘŚĆ 6. Tworzenie interfejsów użytkownika końcowego na podstawie	911
ROZDZIAŁ 27. Wprowadzenie do WPF (Windows Presentation Foundation) i XAML	913
Motywacja twórców WPF	913
Różne odmiany aplikacji WPF	916
Analiza pakietów WPF	920
Tworzenie aplikacji WPF bez XAML	926

Tworzenie aplikacji WPF w samym XAML	934
Transformacja znakowania na pakiet .NET	938
Składnia WPF XAML	942
Tworzenie aplikacji WPF za pomocą plików code-behind	950
Tworzenie aplikacji WPF w Visual Studio 2010	952
Podsumowanie	961
ROZDZIAŁ 28. Programowanie z wykorzystaniem kontrolek WPF	963
Przegląd podstawowych kontrolek WPF	963
Kontrolowanie rozmieszczenia zawartości za pomocą paneli	967
Tworzenie ramki okna za pomocą zagnieżdżonych paneli	976
Polecenia kontrolne WPF	981
Tworzenie interfejsu użytkownika WPF za pomocą Expression Blend	986
Tworzenie karty Ink API	993
Wprowadzenie do API Documents	1002
Tworzenie karty Documents	1003
Wprowadzenie do modelu dowiązywania danych WPF	1009
Podsumowanie	1016
ROZDZIAŁ 29. Renderowanie grafiki WPF	1017
Omówienie renderowania grafiki WPF	1017
Renderowanie danych graficznych za pomocą kształtów	1019
Pędzle i pióra WPF	1027
Stosowanie transformacji graficznych	1031
Używanie kształtów za pomocą Expression Blend	1035
Renderowanie danych graficznych za pomocą rysowania i geometrii	1038
Generowanie złożonej grafiki wektorowej w programie Expression Design	1041
Renderowanie danych graficznych za pomocą warstwy wizualnej	1043
Podsumowanie	1049
ROZDZIAŁ 30. Zasoby, animacje i style WPF	1051
System zasobów WPF	1051
Używanie zasobów obiektowych (logicznych)	1056
Animacja WPF	1065
Tworzenie animacji w języku XAML	1070
Rola stylów WPF	1073
Generowanie stylów w programie Expression Blend	1079
Podsumowanie	1082
ROZDZIAŁ 31. Szablony kontrolek WPF i kontrolki użytkownika	1083
Rola właściwości zależnościowych	1083
Tworzenie niestandardowej właściwości zależnościowej	1088
Zdarzenia trasowane	1093
Drzewa logiczne, drzewa wizualne i szablony domyślne	1096
Tworzenie niestandardowego szablonu kontrolki w Visual Studio 2010	1102

Tworzenie niestandardowych kontrolki użytkownika w programie Expression Blend	1108
Tworzenie aplikacji WPF Jackpot Deluxe	1114
Podsumowanie	1123
CZEŚĆ 7. Tworzenie aplikacji sieciowych za pomocą ASP.NET	1125
ROZDZIAŁ 32. Tworzenie internetowych stron ASP.NET	1127
Rola HTTP	1127
Aplikacje sieciowe i serwery WWW	1128
Rola HTML	1130
Rola skryptów wykonywanych po stronie klienta	1134
Odesłania na serwer WWW	1136
Zbiór funkcji API ASP.NET	1137
Tworzenie jednoplikowej strony ASP.NET	1140
Tworzenie strony ASP.NET za pomocą plików code-behind	1148
Witryny ASP.NET i aplikacje sieciowe ASP.NET	1153
Struktura katalogów witryny ASP.NET	1154
Łańcuch dziedziczenia typu Page	1156
Komunikacja z przychodzącym żądaniem HTTP	1157
Komunikacja z wychodzącą odpowiedzią HTTP	1160
Cykl życiowy strony ASP.NET	1161
Rola pliku Web.config	1164
Podsumowanie	1166
ROZDZIAŁ 33. Kontrolki sieciowe, strony wzorcowe i motywy ASP.NET	1167
Natura kontrolki sieciowych	1167
Klasy bazowe Control oraz WebControl	1169
Tworzenie witryny ASP.NET o samochodach	1177
Rola kontrolki walidacyjnych	1190
Używanie motywów	1195
Podsumowanie	1200
ROZDZIAŁ 34. Techniki zarządzania stanem ASP.NET	1203
Problemy ze stanem	1203
Techniki zarządzania stanem ASP.NET	1205
Rola stanu widoku ASP.NET	1205
Rola pliku Global.asax	1208
Różnica między aplikacją a sesją	1211
Używanie bufora aplikacji	1215
Zachowywanie danych sesji	1219
Pliki cookie	1222
Rola elementu	1224
API Profile w ASP.NET	1226
Podsumowanie	1232
ZAŁĄCZNIK A. Programowanie z wykorzystaniem Windows Forms	1233

Przestrzenie nazw Windows Forms	1234
Tworzenie prostej aplikacji Windows Forms	1234
Szablon projektu Windows Forms w Visual Studio	1239
Anatomia formatki	1244
Reagowanie na działania myszy i klawiatury	1251
Projektowanie okien dialogowych	1254
Renderowanie danych graficznych za pomocą GDI+	1260
Tworzenie kompletnej aplikacji Windows Forms	1265
Podsumowanie	1271
ZAŁĄCZNIK B. Mono: tworzenie aplikacji .NET niezależnie od platformy systemowej	1273
Niezależność .NET od systemu operacyjnego	1273
Pobieranie i instalowanie Mono	1276
Języki programowania na platformie Mono	1278
Narzędzia programistyczne Mono kompatybilne z produktami Microsoftu	1279
Tworzenie aplikacji .NET w Mono	1281
Kto używa Mono?	1287
Zalecana lektura	1288
Podsumowanie	1289
Indeks	1291