

Spis treści

| | |
|--|-----------|
| Wstęp | 9 |
| O czym jest książka? | 9 |
| Ale czy na pewno przeciętna osoba bez wrodzonych cech programisty może się nauczyć programować? | 11 |
| Struktura książki | 11 |
| | |
| Lekcja 1. Programowanie w C# i Unity – szybki start | 13 |
| Instrukcja instalacji programów Unity i Visual Studio | 13 |
| Unity i Visual Studio – pierwsze uruchomienie i utworzenie pierwszego projektu | 17 |
| Informacje o Unity i Visual Studio | 17 |
| Utworzenie pierwszego projektu | 17 |
| Główne okna Unity | 18 |
| Uruchomienie Visual Studio | 19 |
| Okna Unity oraz wstawienie pierwszych obiektów na scenę | 22 |
| Podstawowe okna Unity | 22 |
| Zasoby i prefabrykaty | 23 |
| Wstawienie obiektów na scenę | 23 |
| Modyfikacja obiektów poprzez okno inspektora | 24 |
| Uruchomienie gry | 25 |
| Podstawowe komponenty: Transform, Collider, Renderer i Rigidbody | 25 |
| Komponenty ogólnie | 25 |
| Komponent Transform | 26 |
| Komponent Renderer | 27 |
| Komponent Collider | 27 |
| Komponent Rigidbody | 27 |
| Pierwszy własny skrypt C# i pierwszy własny komponent | 28 |
| Dodanie własnego komponentu | 28 |
| Kod źródłowy komponentu | 28 |
| Komentarze w kodzie programu | 29 |
| Dodanie parametru do komponentu | 29 |
| Komponent a klasa | 30 |
| Funkcje. Specjalne funkcje komponentów Unity. Użycie funkcji OnCollisionEnter | 31 |
| Obiekty i pobieranie komponentów jako obiektów. Zmiana parametrów i pól obiektu | 32 |
| Bardziej zaawansowany sposób wykonania zadania (opcjonalnie) | 32 |
| Kompilacja projektu w Unity | 34 |
| Porady i pomoc w poruszaniu się po programach Unity i Visual Studio | 35 |
| Mam otwarte Unity, ale nie widać sceny i obiektów | 35 |
| Nie umiem otworzyć Visual Studio tak, aby otwierało mój projekt | |

| | |
|--|-----------|
| i moje pliki źródłowe | 35 |
| Zadania do samodzielnego wykonania | 36 |
| Zadanie 1. Rozbudowa projektu ObjectClicker — część I | 36 |
| Zadanie 2. Rozbudowa projektu ObjectClicker — część II | 37 |
| Zadanie 3. Rozbudowa projektu ObjectClicker — część III | 38 |
| Zadanie 4. Rozbudowa projektu ObjectClicker — część IV | 40 |
| Lekcja 2. Informacja, algorytm, kompilacja | 45 |
| Informacja a algorytm | 45 |
| W świecie komputerów informacja jest wszystkim | 45 |
| W świecie komputerów programista jest najważniejszy | 46 |
| Sposób zapisu informacji to też informacja | 46 |
| Algorytm. Główna rola programisty polega na zapisaniu algorytmu w języku programowania | 46 |
| Do przemyślenia | 47 |
| Algorytmy | 47 |
| Algorytm Pieczenie ciasta | 47 |
| Algorytm obliczania pola powierzchni koła | 50 |
| Zapis algorytmu w schemacie blokowym na przykładzie algorytmu gry 3-5-8 | 51 |
| Zapis algorytmu gry 3-5-8 w pseudokodzie | 53 |
| Dokumentacja oraz procesy tworzenia gier | 53 |
| Metody tworzenia gier i programów | 53 |
| Role w projekcie | 54 |
| Dokumentacja projektowa | 56 |
| Kompilacja projektu w Unity i Visual Studio, biblioteki, pliki projektu | 58 |
| Kompilacja projektu w Unity | 58 |
| Pliki bibliotek | 59 |
| Pliki projektu i rozwiązania | 60 |
| Inne pliki i katalogi | 61 |
| Kompilacja kodu źródłowego w Visual Studio | 61 |
| Zadania do samodzielnego wykonania | 62 |
| Samodzielne utworzenie dokumentacji projektowej gry | 62 |
| Lekcja 3. Typy danych, zmienne, funkcje i klasy | 63 |
| Jak manipulować obiektami na scenie 3D w Unity | 63 |
| Przybornik narzędzi | 63 |
| Przyciąganie podczas przesuwania | 64 |
| Szybkie debugowanie zmiennych w Unity i VS | 64 |
| Typy danych i zmienne | 66 |
| Typy danych i zmienne | 66 |
| Rzutowanie zmiennych | 69 |
| Funkcje | 69 |
| Klasy — część I | 72 |
| Lekcja 4. Instrukcje warunkowe | 75 |
| Instrukcje warunkowe — część I | 75 |
| Zadania do samodzielnego wykonania | 77 |

| | |
|-------------------------------------|----|
| Projekt JakimJestemTypem — część I | 77 |
| ProjektRosliny — część I | 79 |
| ProjektRosliny — część II | 81 |
| Projekt JakimJestemTypem — część II | 85 |

**Lekcja 5. Typy danych, klasy, instrukcje warunkowe
— ciąg dalszy. Programowanie w Unity 89**

| | |
|--|-----|
| Klasy — część II | 89 |
| Dziedziczenie klas | 89 |
| Tworzenie nowych obiektów | 92 |
| Złożenia wywołań metod i pól | 92 |
| Słowo kluczowe this | 93 |
| Rzutowanie typów obiektowych | 94 |
| Instrukcje warunkowe — część II | 95 |
| Zagnieżdżanie warunków | 95 |
| Kolejność obliczeń | 96 |
| Operator ! | 96 |
| Zadanie do samodzielnego wykonania | 97 |
| Projekt JakimJestemTypem — część III | 97 |
| Programowanie komponentów Unity | 100 |
| Funkcje Update i OnGUI | 100 |
| Zmiana pozycji obiektu sceny w hierarchii obiektów | 102 |
| Aktywność obiektu | 103 |
| GameObject.Find | 103 |
| Zadania do samodzielnego wykonania | 104 |
| ProjektRosliny — część III | 104 |
| Gra 3-5-8 | 108 |

Lekcja 6. Kolekcje i pętle 113

| | |
|---|-----|
| Kolekcje | 113 |
| Kolekcje ogólnie | 113 |
| Deklaracja i używanie tablic | 113 |
| Ćwiczenie | 114 |
| Pętle for i foreach | 115 |
| Pętla for | 115 |
| Przykłady pętli for | 116 |
| Zadanie do samodzielnego wykonania | 119 |
| Ćwiczenie pętli for | 119 |
| Pętla foreach | 119 |
| Zadania do samodzielnego wykonania | 121 |
| Ćwiczenie pętli foreach | 121 |
| Projekt ObjectClicker— część V | 122 |
| Pętla while | 123 |
| Pętla while | 123 |
| Zadanie do samodzielnego wykonania | 125 |
| Projekt ObjectClicker — część VI — wersja z pętlą while | 125 |
| Podsumowanie tematyki pętli | 125 |
| Zadania do samodzielnego wykonania | 126 |

| | |
|---|------------|
| ProjektRosliny — część IV | 126 |
| Gra MicroAbrix | 130 |
| Lekcja 7. Cykl życia obiektów, przestrzenie nazw, instrukcja switch, typ string, debugowanie | 133 |
| Konstruktory i przeciążanie metod | 133 |
| Konstruktory. Cykl życia obiektów | 133 |
| Przeciążanie metod | 134 |
| Widoczność zmiennych | 136 |
| Widoczność zmiennych wewnątrz bloków | 136 |
| Pola klasy | 137 |
| Inne sposoby przekazywania parametrów: out i ref | 139 |
| Pola statyczne klasy | 141 |
| Przestrzenie nazw | 142 |
| Instrukcja switch | 143 |
| Instrukcja switch | 143 |
| Instrukcje break i continue | 144 |
| Typ string | 144 |
| Typy danych string i char | 144 |
| Metody i pola klasy String | 145 |
| Klasa StringBuilder | 145 |
| Konwersja liczb na tekst i odwrotnie | 146 |
| Debugowanie w Visual Studio | 146 |
| Podłączenie Visual Studio do gry uruchomionej w Unity | 146 |
| Stos wywołań | 148 |
| Podgląd i modyfikacja wartości zmiennych | 148 |
| Kontynuacja wykonania programu po pułapce | 149 |
| Zadania do samodzielnego wykonania | 150 |
| Projekt FixMe1 | 150 |
| Projekt FixMe2 | 152 |
| Uszkodzony ProjektRosliny — część V | 155 |
| Lekcja 8. Typ enum. Użycie gotowego kodu | 161 |
| Typ wyliczeniowy — Enum | 161 |
| Typ enum | 161 |
| Rzutowanie typu enum | 162 |
| Wyszukiwanie i użycie dokumentacji technicznej. Wykorzystanie gotowego kodu | 163 |
| Pomoc techniczna Unity i Visual Studio | 163 |
| Uzyskiwanie pomocy w internecie | 165 |
| Zakup zasobów. Unity Asset Store | 165 |
| Wady i zalety używania gotowego kodu | 167 |
| Zadania do samodzielnego wykonania | 169 |
| Gotowy projekt Tanks! | 169 |
| Skorowidz | 171 |