

# Spis treści

ELEMENTY KSIĄŻKI (16)

PRZEDMOWA (23)

## 1. WPROWADZENIE (35)

1.1. Czym jest programowanie? (35)

1.2. Anatomia komputera (36)

IAS: Komputery są wszędzie (38)

1.3. Kod maszynowy i języki programowania (39)

IAS: Organizacje normalizacyjne (40)

1.4. Zapoznanie się ze środowiskiem programowania (41)

WDP: Kopie zapasowe (44)

1.5. Analiza pierwszego programu (45)

CPB: Pomijanie średników (47)

TS: Sekwencje ucieczki (48)

1.6. Błędy (49)

CPB: Błędna pisownia wyrazów (50)

1.7. Rozwiązywanie problemów: projektowanie algorytmów (50)

1.7.1. Koncepcja algorytmu (51)

1.7.2. Algorytm rozwiązywania problemu stopy zwrotu (51)

1.7.3. Pseudokod (52)

1.7.4. Od algorytmów do programów (53)

JTZ: Opisywanie algorytmu za pomocą pseudokodu (54)

P: Napisanie algorytmu układania płytek podłogowych (55)

Podsumowanie rozdziału (57)

## 2. PODSTAWOWE TYPY DANYCH (59)

2.1. Zmienne (59)

- 2.1.1. Definicje zmiennych (60)
- 2.1.2. Typy liczbowe (61)
- 2.1.3. Nazwy zmiennych (62)
- 2.1.4. Instrukcja przypisania (63)
- 2.1.5. Stałe (65)
- 2.1.6. Komentarze (65)
  - CPB: Używanie niezdefiniowanych zmiennych (66)
  - CPB: Używanie niezainicjowanych zmiennych (66)
  - WDP: Wybieraj opisowe nazwy zmiennych (67)
  - WDP: Nie używaj sekretnych numerów (67)
  - TS: Typy liczbowe w C++ (68)
  - TS: Zakresy i precyzja liczb (69)
  - TS: Definiowanie zmiennych ze słowem auto (69)
- 2.2. Arytmetyka (69)
  - 2.2.1. Operatory arytmetyczne (69)
  - 2.2.2. Inkrementacja i dekrementacja (70)
  - 2.2.3. Dzielenie całkowite i reszta z dzielenia (70)
  - 2.2.4. Konwertowanie liczb zmiennoprzecinkowych na całkowite (71)
  - 2.2.5. Potęgi i pierwiastki (72)
    - CPB: Niezamierzone dzielenie całkowite (74)
    - CPB: Niezamknięte nawiasy (74)
    - CPB: Zapominanie o plikach nagłówkowych (75)
    - CPB: Błędy zaokrąglenia (76)
    - WDP: Spacje w wyrażeniach (76)
    - TS: Rzutowania (77)
    - TS: Połączenie przypisania i działań arytmetycznych (77)
    - IAS: Błąd jednostki zmiennoprzecinkowej procesora Pentium (77)
- 2.3. Wejście i wyjście (79)
  - 2.3.1. Wejście (79)
  - 2.3.2. Formatowanie wyjścia (80)
- 2.4. Rozwiązywanie problemów: najpierw zrób to ręcznie (82)

P: Obliczanie czasu podróży (83)

JTZ: Przeprowadzanie obliczeń (83)

P: Obliczenie kosztu znaczków pocztowych (86)

## 2.5. Ciągi (86)

2.5.1. Typ string (86)

2.5.2. Łączenie ciągów (87)

2.5.3. Wprowadzanie ciągów (87)

2.5.4. Funkcje ciągów (88)

IAS: Alfabet międzynarodowe i zestaw Unicode (91)

Podsumowanie rozdziału (92)

## 3. DECYZJE (93)

### 3.1. Instrukcja if (93)

CPB: Średnik po warunku if (96)

WDP: Układ nawiasów klamrowych (96)

WDP: Zawsze używaj nawiasów klamrowych (97)

WDP: Wcięcia (97)

WDP: Unikaj duplikowania kodu w gałęziach (98)

TS: Operator warunkowy (99)

### 3.2. Porównywanie liczb oraz znaków (99)

CPB: Mylenie operatora = z == (101)

CPB: Dokładne porównywanie liczb zmiennoprzecinkowych (102)

WDP: Doprowadź do kompilacji z brakiem ostrzeżeń (103)

TS: Alfabetyczna kolejność ciągów (103)

JTZ: Implementacja instrukcji if (104)

P: Wyodrębnianie ze środka (106)

IAS: Dysfunkcjonalne systemy komputerowe (106)

### 3.3. Wiele wariantów (107)

TS: Instrukcja switch (110)

### 3.4. Zagnieżdżone gałęzie (111)

CPB: Problem z zawieszonym słowem else (113)

WDP: Ręczne śledzenie kodu (114)

3.5. Rozwiązywanie problemów: schematy blokowe (116)

3.6. Rozwiązywanie problemów: przypadki testowe (119)

WDP: Opracuj harmonogram i zarezerwuj czas na nieoczekiwane problemy (120)

3.7. Zmienne i operatory logiczne (121)

CPB: Łączenie wielu operatorów relacyjnych (124)

CPB: Mylenie warunków && i || (125)

TS: Skrócone obliczanie wartości operatorów logicznych (126)

TS: Prawa de Morgana (126)

3.8. Zastosowanie: weryfikacja danych wejściowych (127)

IAS: Sztuczna inteligencja (129)

Podsumowanie rozdziału (131)

#### 4. PĘTLE (133)

4.1. Pętla while (133)

CPB: Nieskończone pętle (138)

CPB: Nie myśl w kategoriach "Czy doszliśmy już do celu?" (138)

CPB: Pomyłki o jeden (139)

IAS: Pierwszy "bug" (140)

4.2. Rozwiązywanie problemów: ręczne śledzenie kodu (140)

4.3. Pętla for (143)

WDP: Używaj pętli tylko do tego, do czego została przeznaczona (147)

WDP: Wybierz zakres pętli odpowiedni do zadania (147)

WDP: Licz iteracje (148)

4.4. Pętla do (148)

WDP: Schematy blokowe pętli (149)

4.5. Przetwarzanie danych wejściowych (150)

4.5.1. Wartości wartownika (150)

4.5.2. Odczytywanie danych wejściowych do chwili niepowodzenia (151)

TS: Czyszczenie stanu błędu (153)

TS: "Pętla i pół" oraz instrukcja break (154)

- TS: Przekierowywanie wejścia i wyjścia (154)
- 4.6. Rozwiązywanie problemów: scenopis (155)
- 4.7. Typowe algorytmy pętli (158)
  - 4.7.1. Suma i średnia (158)
  - 4.7.2. Zliczanie pasujących elementów (158)
  - 4.7.3. Znajdowanie pierwszego pasującego elementu (159)
  - 4.7.4. Monitowanie aż do skutku (159)
  - 4.7.5. Maksimum i minimum (160)
  - 4.7.6. Porównywanie sąsiednich wartości (160)
- JTZ: Tworzenie pętli (161)
- P: Przetwarzanie numerów kart kredytowych (165)
- 4.8. Zagnieżdżone pętle (165)
  - P: Manipulowanie pikselami obrazu (168)
- 4.9. Rozwiązywanie problemów: najpierw rozwiąż prostszy problem (169)
- 4.10. Liczby losowe i symulacje (173)
  - 4.10.1. Generowanie liczb losowych (174)
  - 4.10.2. Symulowanie rzutów kostką (175)
  - 4.10.3. Metoda Monte Carlo (176)
- IAS: Piractwo cyfrowe (177)
- Podsumowanie rozdziału (178)
  
- 5. FUNKCJE (181)
  - 5.1. Funkcje jako czarne skrzynki (181)
  - 5.2. Implementowanie funkcji (183)
    - WDP: Komentarze funkcji (185)
  - 5.3. Przekazywanie parametrów (185)
    - WDP: Nie modyfikuj zmiennych parametrycznych (187)
  - 5.4. Wartości zwracane (187)
    - CPB: Brak wartości zwracanej (188)
    - TS: Deklaracje funkcji (189)
    - JTZ: Implementowanie funkcji (190)

P: Generowanie losowych haseł (191)

P: Używanie debugera (191)

5.5. Funkcje bez wartości zwracanych (192)

5.6. Rozwiązywanie problemów: funkcje do ponownego wykorzystania (193)

5.7. Rozwiązywanie problemów: uściślanie stopniowe (195)

WDP: Pilnuj, by funkcje były krótkie (200)

WDP: Śledzenie funkcji (200)

WDP: Atrapy (202)

P: Obliczanie oceny z przedmiotu (202)

5.8. Zakres zmiennej i zmienne globalne (202)

WDP: Unikaj zmiennych globalnych (204)

5.9. Parametry referencyjne (204)

WDP: Preferuj wartości zwracane zamiast parametrów referencyjnych (208)

TS: Stałe referencje (209)

5.10. Funkcje rekurencyjne (opcjonalnie) (209)

JTZ: Wnioskowanie rekurencyjne (212)

IAS: Nagłe rozpowszechnienie się komputerów osobistych (214)

Podsumowanie rozdziału (215)

6. TABLICE I WEKTORY (217)

6.1. Tablice (217)

6.1.1. Definiowanie tablic (217)

6.1.2. Dostęp do elementów tablicy (219)

6.1.3. Częściowo wypełnione tablice (221)

CPB: Przekroczenie zakresu (222)

WDP: Używaj tablic do przechowywania serii związanych ze sobą wartości (222)

IAS: Wirusy komputerowe (222)

6.2. Typowe algorytmy tablicowe (224)

6.2.1. Wypełnianie wartościami (224)

6.2.2. Kopiowanie (224)

6.2.3. Suma i średnia (225)

- 6.2.4. Maksimum i minimum (225)
- 6.2.5. Separatory elementów (225)
- 6.2.6. Zliczanie pasujących elementów (226)
- 6.2.7. Wyszukiwanie liniowe (226)
- 6.2.8. Usuwanie elementu (227)
- 6.2.9. Wstawianie elementu (227)
- 6.2.10. Przeszycanie elementów (229)
- 6.2.11. Odczyt danych wejściowych (230)
  - TS: Sortowanie za pomocą biblioteki C++ (231)
  - TS: Algorytm sortowania (231)
  - TS: Wyszukiwanie binarne (233)
- 6.3. Tablice a funkcje (234)
  - TS: Stałe parametry tablicowe (237)
- 6.4. Rozwiązywanie problemów: dostosowywanie algorytmów (238)
  - JTZ: Praca z tablicami (240)
  - P: Rzut kostką (243)
- 6.5. Rozwiązywanie problemów: odkrywanie algorytmów przez manipulację obiektami fizycznymi (244)
- 6.6. Tablice dwuwymiarowe (246)
  - 6.6.1. Definiowanie tablic dwuwymiarowych (247)
  - 6.6.2. Dostęp do elementów (248)
  - 6.6.3. Lokalizowanie sąsiadujących elementów (248)
  - 6.6.4. Obliczanie sum wierszy i kolumn (249)
  - 6.6.5. Dwuwymiarowe parametry tablicowe (250)
    - CPB: Pomijanie rozmiaru kolumny w dwuwymiarowym parametrze tablicowym (253)
    - P: Tabela danych o ludności świata (253)
- 6.7. Wektory (253)
  - 6.7.1. Definiowanie wektorów (254)
  - 6.7.2. Powiększanie i zmniejszanie wektorów (255)
  - 6.7.3. Wektory a funkcje (256)
  - 6.7.4. Algorytmy związane z wektorami (257)

#### 6.7.5. Wektory dwuwymiarowe (259)

WDP: Stosuj wektory zamiast tablic (260)

TS: Pętla for oparta na zakresie (260)

Podsumowanie rozdziału (261)

### 7. WSKAŹNIKI I STRUKTURY (263)

#### 7.1. Definiowanie i używanie wskaźników (264)

7.1.1. Definiowanie wskaźników (264)

7.1.2. Dostęp do zmiennych poprzez wskaźniki (265)

7.1.3. Inicjowanie wskaźników (266)

CPB: Mylenie wskaźników z danymi, na które wskazują (268)

WDP: Używaj oddzielnej definicji dla każdej zmiennej wskaźnikowej (269)

TS: Wskaźniki i referencje (269)

#### 7.2. Tablice i wskaźniki (270)

7.2.1. Tablice jako wskaźniki (270)

7.2.2. Arytmetyka wskaźnikowa (271)

7.2.3. Tablicowe zmienne parametryczne są wskaźnikami (272)

TS: Przechodzenie po tablicy przy użyciu wskaźnika (273)

CPB: Zwracanie wskaźnika wskazującego na zmienną lokalną (274)

WDP: Programuj przejrzystość, a nie sprytnie (275)

TS: Stałe wskaźniki (275)

#### 7.3. Ciągi w językach C i C++ (276)

7.3.1. Typ char (276)

7.3.2. Ciągi w stylu C (276)

7.3.3. Tablice znaków (277)

7.3.4. Konwertowanie pomiędzy ciągami w stylu C i C++ (278)

7.3.5. Ciągi w stylu C++ i operator [] (278)

TS: Praca z ciągami w stylu C (279)

#### 7.4. Dynamiczna alokacja pamięci (281)

CPB: Wiszące wskaźniki (283)

CPB: Wycieki pamięci (284)

7.5. Tablice i wektory wskaźników (285)

7.6. Rozwiązywanie problemów: rysowanie schematu (288)

JTZ: Praca ze wskaźnikami (289)

P: Tworzenie korespondencji masowej (291)

IAS: Systemy wbudowane (291)

7.7. Struktury (292)

7.7.1. Typy strukturalne (292)

7.7.2. Przypisania struktur i ich porównywanie (293)

7.7.3. Funkcje a struktury (294)

7.7.4. Tablice struktur (294)

7.7.5. Struktury ze składowymi tablicowymi (295)

7.7.6. Struktury zagnieżdżone (295)

7.8. Wskaźniki a struktury (296)

7.8.1. Wskaźniki do struktur (296)

7.8.2. Struktury ze składowymi wskaźnikowymi (297)

TS: Wskaźniki inteligentne (298)

Podsumowanie rozdziału (299)

8. STRUMIENIE (301)

8.1. Odczytywanie i zapisywanie plików tekstowych (301)

8.1.1. Otwieranie strumienia (302)

8.1.2. Odczyt z pliku (303)

8.1.3. Zapis do pliku (304)

8.1.4. Przykład przetwarzania pliku (304)

8.2. Odczyt tekstowych danych wejściowych (307)

8.2.1. Odczyt wyrazów (307)

8.2.2. Odczyt znaków (307)

8.2.3. Odczyt wierszy (309)

CPB: Łączenie operacji wejścia przy użyciu operatora >> i funkcji getline (310)

TS: Sprawdzanie błędu strumienia (311)

8.3. Zapisywanie tekstowych danych wyjściowych (312)

TS: Standard Unicode, kodowanie UTF-8 i ciągi C++ (314)

8.4. Analizowanie i formatowanie ciągów (315)

8.5. Argumenty wiersza poleceń (317)

IAS: Algorytmy szyfrowania (320)

JTZ: Przetwarzanie plików tekstowych (321)

P: Wyszukiwanie duplikatów (324)

8.6. Dostęp swobodny i pliki binarne (324)

8.6.1. Dostęp swobodny (324)

8.6.2. Pliki binarne (325)

8.6.3. Przetwarzanie plików z obrazami (326)

IAS: Bazy danych a prywatność (329)

Podsumowanie rozdziału (330)

9. KLASY (333)

9.1. Programowanie obiektowe (334)

9.2. Implementowanie prostej klasy (335)

9.3. Określanie interfejsu publicznego klasy (338)

CPB: Zapominanie o średniku (340)

9.4. Projektowanie reprezentacji danych (341)

9.5. Funkcje składowe (342)

9.5.1. Implementowanie funkcji składowych (342)

9.5.2. Parametry jawne i niejawne (343)

9.5.3. Wywoływanie funkcji składowej wewnątrz innej funkcji składowej (344)

WDP: Wszystkie dane składowe powinny być prywatne, a większość funkcji - publiczna (347)

WDP: Poprawne użycie słowa zastrzeżonego const (347)

9.6. Konstruktory (348)

CPB: Próba wywołania konstruktora (350)

TS: Przeciążanie (351)

TS: Listy inicjatorów (351)

TS: Uniwersalna i jednolita składnia inicjacji (352)

9.7. Rozwiązywanie problemów: śledzenie obiektów (353)

JTZ: Implementowanie klasy (355)

P: Implementowanie klasy reprezentującej konto bankowe (358)

IAS: Elektroniczne maszyny do głosowania (358)

#### 9.8. Rozwiązywanie problemów: znajdowanie klas (360)

WDP: Przekształcaj wektory równoległe w wektory obiektów (361)

#### 9.9. Osobna kompilacja (363)

#### 9.10. Wskaźniki do obiektów (367)

9.10.1. Obiekty alokowane dynamicznie (367)

9.10.2. Operator -> (368)

9.10.3. Wskaźnik this (369)

#### 9.11. Rozwiązywanie problemów: wzorce danych obiektu (369)

9.11.1. Przechowywanie sumy całkowitej (370)

9.11.2. Liczenie zdarzeń (371)

9.11.3. Gromadzenie danych (371)

9.11.4. Zarządzanie właściwościami obiektu (372)

9.11.5. Modelowanie obiektów o różnych stanach (373)

9.11.6. Opisywanie pozycji obiektu (374)

IAS: Oprogramowanie o otwartych źródłach i wolne oprogramowanie (375)

Podsumowanie rozdziału (376)

### 10. DZIEDZICZENIE (379)

10.1. Hierarchie dziedziczenia (379)

10.2. Implementowanie klas pochodnych (383)

CPB: Dziedziczenie prywatne (386)

CPB: Replikowanie składowych klasy bazowej (386)

WDP: Przy odmiennych wartościach należy używać jednej klasy, przy odmiennym działaniu - dziedziczenia (387)

TS: Wywoływanie konstruktora klasy bazowej (387)

10.3. Przesłanianie funkcji składowych (388)

CPB: Zapominanie o podaniu nazwy klasy bazowej (391)

10.4. Funkcje wirtualne i polimorfizm (391)

10.4.1. Problem odcinania (392)

10.4.2. Wskaźniki do klasy bazowej i pochodnej (393)

10.4.3. Funkcje wirtualne (394)

10.4.4. Polimorfizm (395)

WD: Nie używaj znaczników typów (398)

CPB: Odcinanie obiektu (398)

CPB: Nieudane przesłanianie funkcji wirtualnej (399)

TS: Wirtualne samowywołania (400)

JTZ: Opracowywanie hierarchii dziedziczenia (400)

P: Implementowanie hierarchii pracowników a potrzeby przetwarzania odcinków wypłat (406)

IAS: Kto sprawuje kontrolę nad internetem? (406)

Podsumowanie rozdziału (408)

## 11. REKURENCJA (409)

11.1. Liczby trójkątne (409)

CPB: Śledzenie wykonania funkcji rekurencyjnych (413)

CPB: Nieskończona rekurencja (414)

JTZ: Wnioskowanie rekurencyjne (415)

P: Znajdowanie plików (418)

11.2. Rekurencyjne funkcje pomocnicze (418)

11.3. Wydajność rekurencji (419)

11.4. Permutacje (423)

11.5. Rekurencja wzajemna (426)

11.6. Poszukiwanie z nawrotami (430)

P: Wieże Hanoi (436)

IAS: Ograniczenia obliczeń komputerowych (436)

Podsumowanie rozdziału (439)

## 12. SORTOWANIE I WYSZUKIWANIE (441)

12.1. Sortowanie przez wybieranie (441)

- 12.2. Profilowanie algorytmu sortowania przez wybieranie (444)
- 12.3. Analiza wydajności algorytmu sortowania przez wybieranie (445)
  - TS: O, omega i theta (447)
  - TS: Sortowanie przez wstawianie (449)
- 12.4. Sortowanie przez scalanie (450)
- 12.5. Analiza algorytmu sortowania przez scalanie (453)
  - TS: Algorytm sortowania szybkiego (456)
- 12.6. Wyszukiwanie (457)
  - 12.6.1. Wyszukiwanie liniowe (457)
  - 12.6.2. Wyszukiwanie binarne (459)
    - WDP: Funkcje biblioteczne do sortowania i wyszukiwania binarnego (462)
    - TS: Definiowanie kolejności sortowania obiektów (462)
- 12.7. Rozwiązywanie problemów: szacowanie czasu wykonania algorytmu (463)
  - 12.7.1. Czas liniowy (463)
  - 12.7.2. Czas kwadratowy (464)
  - 12.7.3. Wzór trójkąta (465)
  - 12.7.4. Czas logarytmiczny (467)
    - P: Ulepszanie algorytmu sortowania przez wstawianie (468)
    - IAS: Pierwsza programistka (468)
- Podsumowanie rozdziału (469)

## 13. ZAAWANSOWANE CECHY JĘZYKA C++ (471)

- 13.1. Przeciążanie operatorów (471)
  - 13.1.1. Funkcje operatorów (472)
  - 13.1.2. Przeciążanie operatorów porównania (474)
  - 13.1.3. Wejście i wyjście (475)
  - 13.1.4. Operatory składowe (476)
    - TS: Przeciążanie operatorów inkrementacji i dekrementacji (476)
    - TS: Niejawne konwersje typów (478)
    - TS: Zwracanie referencji (479)
    - P: Klasa Fraction (480)

## 13.2. Automatyczne zarządzanie pamięcią (480)

### 13.2.1. Konstruktory przydzielające pamięć (480)

### 13.2.2. Destruktory (482)

### 13.2.3. Przeciążanie operatora przypisania (483)

### 13.2.4. Konstruktory kopiujące (488)

WDP: Używaj parametrów referencyjnych, by nie tworzyć kopii obiektów (492)

CPB: Definiowanie destruktora bez pozostałych dwu funkcji z "wielkiej trójki" (492)

TS: Wirtualne destruktory (493)

TS: Powstrzymanie automatycznego tworzenia funkcji zarządzających pamięcią (494)

TS: Operacje przenoszenia (494)

TS: Wskaźniki współdzielone (496)

P: Śledzenie zarządzania pamięcią w obiektach typu String (496)

## 13.3. Szablony (497)

### 13.3.1. Szablony funkcji (497)

### 13.3.2. Szablony klas (499)

TS: Parametry szablonów niedotyczące typów (501)

## Podsumowanie rozdziału (502)

## 14. LISTY POWIĄZANE, STOSY I KOLEJKI (503)

### 14.1. Używanie list powiązanych (503)

### 14.2. Implementowanie list powiązanych (509)

#### 14.2.1. Klasy dla list, węzłów i iteratorów (509)

#### 14.2.2. Implementowanie iteratorów (511)

#### 14.2.3. Implementowanie wstawiania i usuwania węzłów (512)

P: Implementowanie szablonu listy powiązanej (522)

### 14.3. Wydajność operacji na listach, tablicach i wektorach (523)

### 14.4. Stosy i kolejki (527)

### 14.5. Implementowanie stosów i kolejek (530)

#### 14.5.1. Stosy jako listy powiązane (530)

#### 14.5.2. Stosy jako tablice (533)

#### 14.5.3. Kolejki jako listy powiązane (534)

14.5.4. Kolejki jako tablice cykliczne (535)

14.6. Zastosowania stosów i kolejek (536)

14.6.1. Sprawdzanie zamknięcia nawiasów (536)

14.6.2. Obliczanie wyrażeń zapisanych w odwrotnej notacji polskiej (537)

14.6.3. Obliczanie wartości wyrażeń algebraicznych (539)

14.6.4. Poszukiwanie z nawrotami (543)

TS: Odwrotna notacja polska (544)

Podsumowanie rozdziału (545)

15. ZBIORY, MAPY I TABLICE MIESZAJĄCE (547)

15.1. Zbiory (547)

15.2. Mapy (551)

WDP: W przypadku iteratorów używaj typu auto (555)

TS: Multizbiory i multimapy (555)

P: Częstość występowania wyrazów (556)

15.3. Implementowanie tablicy mieszającej (556)

15.3.1. Skrót (556)

15.3.2. Tablice mieszające (558)

15.3.3. Znajdowanie elementu (559)

15.3.4. Dodawanie i usuwanie elementów (560)

15.3.5. Przechodzenie po tablicy mieszającej (560)

TS: Implementowanie funkcji skrótów (566)

TS: Adresowanie otwarte (568)

Podsumowanie rozdziału (570)

16. STRUKTURY DRZEW (571)

16.1. Podstawowe koncepcje dotyczące drzew (571)

16.2. Drzewa binarne (575)

16.2.1. Przykłady drzew binarnych (575)

16.2.2. Drzewa zrównoważone (577)

16.2.3. Implementacja drzewa binarnego (578)

P: Budowanie drzewa Huffmana (580)

### 16.3. Binarne drzewa poszukiwań (580)

16.3.1. Właściwość wyszukiwania binarnego (580)

16.3.2. Wstawianie (582)

16.3.3. Usuwanie (584)

16.3.4. Wydajność operacji (586)

### 16.4. Przeglądanie drzewa (591)

16.4.1. Przeglądanie poprzeczne (591)

16.4.2. Przeglądanie wzdłużne i wsteczne (593)

16.4.3. Wzorzec Wizytator (594)

16.4.4. Przeszukiwanie w głąb i wszere (595)

16.4.5. Iteratory drzew (596)

### 16.5. Drzewa czerwono-czarne (597)

16.5.1. Podstawowe własności drzew czerwono-czarnych (597)

16.5.2. Wstawianie (600)

16.5.3. Usuwanie (601)

P: Implementowanie drzewa czerwono-czarnego (605)

Podsumowanie rozdziału (605)

## 17. KOLEJKI PRIORYTETOWE I KOPCE (607)

17.1. Kolejki priorytetowe (607)

P: Symulacja kolejki oczekujących klientów (610)

17.2. Kopce (610)

17.3. Algorytm sortowania przez kopcowanie (621)

Podsumowanie rozdziału (626)

## A ZESTAWIENIE SŁÓW ZASTRZEŻONYCH (627)

## B ZESTAWIENIE OPERATORÓW (631)

## C KODY ZNAKÓW (633)

D PRZEGLĄD BIBLIOTEKI C++ (637)

E WYTYCZNE DOTYCZĄCE PROGRAMOWANIA W JĘZYKU C++ (643)

F SYSTEMY LICZBOWE (651)

SŁOWNICZEK (661)

ŹRÓDŁA ILUSTRACJI (671)

ŚCIAĞAWKA (673)