

Wstęp (11)

Rozdział 1. Programowanie obiektowe w C++ (15)

- 1.1. Abstrakcyjne typy danych (15)
- 1.2. Enkapsulacja (15)
- 1.3. Dziedziczenie (19)
- 1.4. Wskaźniki (22)
 - 1.4.1. Wskaźniki a tablice (24)
 - 1.4.2. Wskaźniki a konstruktory kopiujące (26)
 - 1.4.3. Wskaźniki i destruktory (29)
 - 1.4.4. Wskaźniki a referencje (29)
 - 1.4.5. Wskaźniki na funkcje (32)
- 1.5. Polimorfizm (33)
- 1.6. C++ a programowanie obiektowe (35)
- 1.7. Standardowa biblioteka szablonów (36)
 - 1.7.1. Kontenery (36)
 - 1.7.2. Iteratory (37)
 - 1.7.3. Algorytmy (37)
 - 1.7.4. Funktory (38)
- 1.8. Wektory w standardowej bibliotece szablonów (40)
- 1.9. Struktury danych a programowanie obiektowe (46)
- 1.10. Przykład zastosowania: plik z dostępem swobodnym (47)
- 1.11. Ćwiczenia (56)
- 1.12. Zadania programistyczne (58)
- Bibliografia (60)

Rozdział 2. Analiza złożoności (61)

- 2.1. Złożoność obliczeniowa i asymptotyczna (61)
- 2.2. O-notacja (62)
- 2.3. Właściwości O-notacji (64)
- 2.4. Notacje Θ i Ω (66)
- 2.5. Możliwe problemy (67)
- 2.6. Przykłady złożoności (67)
- 2.7. Określanie złożoności asymptotycznej. Przykłady (68)
- 2.8. Złożoność optymistyczna, średnia i pesymistyczna (71)
- 2.9. Złożoność zamortyzowana (73)
- 2.10. Ćwiczenia (77)
- Bibliografia (80)

Rozdział 3. Listy (81)

- 3.1. Listy jednokierunkowe (81)
 - 3.1.1. Wstawianie (86)
 - 3.1.2. Usuwanie (88)
 - 3.1.3. Wyszukiwanie (93)
- 3.2. Listy dwukierunkowe (93)
- 3.3. Listy cykliczne (97)
- 3.4. Listy z przeskokami (99)
- 3.5. Listy samoorganizujące się (104)

- 3.6. Tablice rzadkie (107)
- 3.7. Listy w bibliotece STL (110)
- 3.8. Kolejki dwustronne w bibliotece STL (113)
- 3.9. Podsumowanie (117)
- 3.10. Przykład zastosowania. Biblioteka (118)
- 3.11. Ćwiczenia (126)
- 3.12. Zadania programistyczne (128)
- Bibliografia (131)

Rozdział 4. Stosy i kolejki (133)

- 4.1. Stosy (133)
- 4.2. Kolejki (140)
- 4.3. Kolejki priorytetowe (147)
- 4.4. Stosy w bibliotece STL (148)
- 4.5. Kolejki w bibliotece STL (148)
- 4.6. Kolejki priorytetowe w bibliotece STL (149)
- 4.7. Przykład zastosowania. Szukanie wyjścia z labiryntu (152)
- 4.8. Ćwiczenia (156)
- 4.9. Zadania programistyczne (159)
- Bibliografia (160)

Rozdział 5. Rekurencja (161)

- 5.1. Definicje rekurencyjne (161)
- 5.2. Wywołania funkcji a implementacja rekurencji (164)
- 5.3. Anatomia wywołania rekurencyjnego (165)
- 5.4. Rekurencja ogonowa (169)
- 5.5. Rekurencja inna niż ogonowa (170)
- 5.6. Rekurencja pośrednia (174)
- 5.7. Rekurencja zagnieżdżona (176)
- 5.8. Nadużywanie rekurencji (177)
- 5.9. Algorytmy z powrotami (180)
- 5.10. Wnioski końcowe (186)
- 5.11. Przykład zastosowania. Rekurencyjny interpreter (187)
- 5.12. Ćwiczenia (194)
- 5.13. Zadania programistyczne (197)
- Bibliografia (199)

Rozdział 6. Drzewa binarne (201)

- 6.1. Drzewa, drzewa binarne i binarne drzewa poszukiwania (201)
- 6.2. Implementacja drzew binarnych (205)
- 6.3. Wyszukiwanie w drzewie binarnym (207)
- 6.4. Przechodzenie po drzewie (210)
 - 6.4.1. Przechodzenie wszerek (210)
 - 6.4.2. Przechodzenie w głąb (211)
 - 6.4.3. Przechodzenie po drzewie w głąb bez stosu (217)
- 6.5. Wstawianie (223)
- 6.6. Usuwanie (225)

- 6.6.1. Usuwanie przez złączanie (226)
 - 6.6.2. Usuwanie przez kopiowanie (229)
- 6.7. Równoważenie drzewa (231)
 - 6.7.1. Algorytm DSW (234)
 - 6.7.2. Drzewa AVL (236)
- 6.8. Drzewa samoorganizujące się (241)
 - 6.8.1. Drzewa samonaprawiające się (241)
 - 6.8.2. Ukosowanie (243)
- 6.9. Sterty (247)
 - 6.9.1. Sterty jako kolejki priorytetowe (249)
 - 6.9.2. Organizowanie tablic w sterzy (251)
- 6.10. Notacja polska i drzewa wyrażeń (255)
 - 6.10.1. Operacje na drzewach wyrażeń (256)
- 6.11. Przykład zastosowania. Zliczanie częstości występowania słów (259)
- 6.12. Ćwiczenia (265)
- 6.13. Zadania programistyczne (268)
- Bibliografia (272)

Rozdział 7. Drzewa wielokierunkowe (275)

- 7.1. Rodzina B-drzew (276)
 - 7.1.1. B-drzewa (277)
 - 7.1.2. B*-drzewa (286)
 - 7.1.3. B+-drzewa (288)
 - 7.1.4. B+-drzewa przedrostkowe (289)
 - 7.1.5. Drzewa bitowe (291)
 - 7.1.6. R-drzewa (294)
 - 7.1.7. 2-4-drzewa (296)
 - 7.1.8. Zbiory i wielozbiory w bibliotece STL (303)
 - 7.1.9. Mapy i multimapy w bibliotece STL (309)
- 7.2. Drzewa słownikowe (313)
- 7.3. Uwagi końcowe (321)
- 7.4. Przykład zastosowania. Sprawdzanie pisowni (321)
- 7.5. Ćwiczenia (330)
- 7.6. Zadania programistyczne (331)
- Bibliografia (334)

Rozdział 8. Grafy (337)

- 8.1. Reprezentacje grafów (338)
- 8.2. Przechodzenie po grafach (340)
- 8.3. Najkrótsza droga (344)
 - 8.3.1. Problem najkrótszych dróg typu "wszystkie-do-wszystkich" (351)
- 8.4. Wykrywanie cykli (353)
 - 8.4.1. Problem znajdowania sumy zbiorów rozłącznych (354)
- 8.5. Drzewa rozpinające (356)
 - 8.5.1. Algorytm Borůvki (357)
 - 8.5.2. Algorytm Kruskala (358)
 - 8.5.3. Algorytm Jarníka-Prima (360)
 - 8.5.4. Metoda Dijkstry (361)

- 8.6. Spójność (361)
 - 8.6.1. Spójność w grafach nieskierowanych (361)
 - 8.6.2. Spójność w grafach skierowanych (364)
- 8.7. Sortowanie topologiczne (365)
- 8.8. Sieci (368)
 - 8.8.1. Przepływy maksymalne (368)
 - 8.8.2. Maksymalne przepływy o minimalnym koszcie (378)
- 8.9. Kojarzenie (383)
 - 8.9.1. Problem przypisania (387)
 - 8.9.2. Skojarzenia w grafach, które nie są dwudzielne (390)
- 8.10. Grafy eulerowskie i hamiltonowskie (392)
 - 8.10.1. Grafy eulerowskie (392)
 - 8.10.2. Grafy hamiltonowskie (393)
- 8.11. Przykład zastosowania. Unikalni reprezentanci (396)
- 8.12. Ćwiczenia (406)
- 8.13. Zadania programistyczne (409)
- Bibliografia (411)

Rozdział 9. Sortowanie (413)

- 9.1. Podstawowe algorytmy sortowania (414)
 - 9.1.1. Sortowanie przez wstawianie (414)
 - 9.1.2. Sortowanie przez wybieranie (417)
 - 9.1.3. Sortowanie bąbelkowe (419)
- 9.2. Drzewa decyzyjne (422)
- 9.3. Efektywne algorytmy sortowania (425)
 - 9.3.1. Sortowanie Shella (425)
 - 9.3.2. Sortowanie przez kopcowanie (429)
 - 9.3.3. Sortowanie szybkie (quicksort) (433)
 - 9.3.4. Sortowanie poprzez scalanie (440)
 - 9.3.5. Sortowanie pozycyjne (443)
- 9.4. Sortowanie przy wykorzystaniu STL (448)
- 9.5. Uwagi końcowe (452)
- 9.6. Przykład zastosowania. Dodawanie wielomianów (452)
- 9.7. Ćwiczenia (459)
- 9.8. Zadania programistyczne (461)
- Bibliografia (463)

Rozdział 10. Mieszanie (465)

- 10.1. Funkcje mieszające (466)
 - 10.1.1. Dzielenie (466)
 - 10.1.2. Składanie (466)
 - 10.1.3. Funkcja "środek kwadratu" (467)
 - 10.1.4. Wycinanie (468)
 - 10.1.5. Zmiana podstawy (468)
- 10.2. Rozwiązywanie problemu kolizji (468)
 - 10.2.1. Adresowanie otwarte (469)
 - 10.2.2. Metoda łańcuchowa (475)
 - 10.2.3. Adresowanie kubełkowe (476)

- 10.3. Usuwanie (477)
- 10.4. Doskonałe funkcje mieszające (479)
 - 10.4.1. Metoda Cichelliego (479)
 - 10.4.2. Algorytm FHCD (482)
- 10.5. Funkcje mieszające dla plików rozszerzalnych (484)
 - 10.5.1. Mieszanie przedłużalne (485)
 - 10.5.2. Mieszanie liniowe (487)
- 10.6. Przykład zastosowania. Mieszanie z użyciem kubełków (490)
- 10.7. Ćwiczenia (498)
- 10.8. Zadania programistyczne (500)
- Bibliografia (501)

Rozdział 11. Kompresja danych (503)

- 11.1. Warunki kompresji danych (503)
- 11.2. Kodowanie Huffmana (505)
 - 11.2.1. Adaptacyjne kodowanie Huffmana (514)
- 11.3. Metoda Shannona-Fano (519)
- 11.4. Kodowanie długości serii (520)
- 11.5. Metoda Ziva-Lempela (521)
- 11.6. Przykład zastosowania. Metoda Huffmana z kodowaniem długości serii (524)
- 11.7. Ćwiczenia (535)
- 11.8. Zadania programistyczne (536)
- Bibliografia (537)

Rozdział 12. Zarządzanie pamięcią (539)

- 12.1. Metody dopasowywania sekwencyjnego (540)
- 12.2. Metody niesekwencyjne (542)
 - 12.2.1. System bliźniaków (543)
- 12.3. Odśmiecanie (551)
 - 12.3.1. Metoda "zaznacz-i-zamiataj" (551)
 - 12.3.2. Metody kopiowania (559)
 - 12.3.3. Krokowe metody odśmiecania (560)
- 12.4. Uwagi końcowe (567)
- 12.5. Przykład zastosowania. Odśmiecanie "w miejscu" (568)
- 12.6. Ćwiczenia (576)
- 12.7. Zadania programistyczne (577)
- Bibliografia (579)

Dodatek A Obliczanie złożoności (583)

- A.1. Szereg harmoniczny (583)
- A.2. Przybliżenie funkcji $\lg(n!)$ (583)
- A.3. Przybliżona średnia złożoność sortowania szybkiego (585)
- A.4. Średnia długość ścieżki w losowych drzewach binarnych (587)

Dodatek B Algorytmy dostępne w STL (589)

- B.1. Algorytmy standardowe (589)

Skorowidz (597)