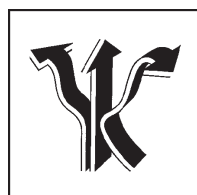


О. Г. ПІСКУНОВ, О. П. ТОМАЩУК,
О. Г. ЧОЛИШКІНА, І. О. ПЕТРЕНКО

РОЗРОБКА КОНСОЛЬНИХ ДОДАТКІВ КОМПІЛЯТОРАМИ C++



МІЖРЕГІОНАЛЬНА
АКАДЕМІЯ УПРАВЛІННЯ ПЕРСОНАЛОМ



МАУП

О. Г. ПІСКУНОВ, О. П. ТОМАЩУК,
О. Г. ЧОЛИШКІНА, І. О. ПЕТРЕНКО

**РОЗРОБКА КОНСОЛЬНИХ ДОДАТКІВ
КОМПІЛЯТОРАМИ
C++**

Підручник

Київ
2023

УДК 004.42(075.8)
Р65

Рецензенти: *А. С. Дуднік*, д-р техн. наук, доц.,
професор кафедри мережевих та інтернет-
технологій Київського національного ун-ту
імені Тараса Шевченка, академік інженерної
академії України

Є. В. Івохін, д-р фіз.-мат. наук, професор кафедри
системного аналізу та теорії прийняття рішень
факультету комп'ютерних наук та кібернетики
Київського національного ун-ту імені Тараса
Шевченка

І. А. Юрчук, канд. фіз.-мат. наук, доцент кафедри
програмних систем і технологій факультету
інформаційних технологій Київського
національного ун-ту імені Тараса Шевченка

*Схвалено Вченою радою Міжрегіональної Академії управління
персоналом (протокол № 8 від 26.10.22)*

Піскунов О. Г.

Р65 Розробка консольних додатків компіляторами С++ :
підручник / О. Г. Піскунов, О. П. Томащук, О. Г. Чолиш-
кіна, І. О. Петренко. Київ : Міжрегіональна Академія
управління персоналом, 2023. 238 с. Бібліогр. : с. 215–219.

ISBN 978-617-02-0383-0

Надано матеріал для вивчення структурного програмування, для
розроблення, тестування і документування консольних додатків на ком-
піляторах С++.

Підручник буде корисним для студентів і аспірантів, які вивчають
програмування, а також усім, хто виявляє інтерес до цієї сфери знань.

УДК 004.42(075.8)

© О. Г. Піскунов, О. П. Томащук,
О. Г. Чолишкіна, І. О. Петренко, 2023
© Міжрегіональна Академія
управління персоналом, 2023

ISBN 978-617-02-0383-0

ЗМІСТ

Вступ	9
Розділ 1. ВВЕДЕННЯ В ПРОГРАМУВАННЯ	10
1.1. МОВИ ПРОГРАМУВАННЯ І АЛГОРИТМИ.....	10
1.1.1. Інтерпретатор cmd і файлова система Windows	14
1.1.1.1. Команди для використання файлів	14
1.1.1.2. Команди для використання каталогів	16
1.1.1.3. Додаткові команди	18
1.1.2. Мова блок-схем	19
1.1.3. Компілятори мови C++	21
1.2. ВІД ПОСТАНОВКИ ЗАДАЧІ ДО СТВОРЕННЯ	
ДОДАТКА	22
1.2.1. Створення першого додатка	22
1.2.2. Основні елементи мови C	27
1.2.3. Введення-виведення у мові C++	30
1.2.4. Кроки розробки програми	31
1.2.5. Блок-схема програми	33
1.2.6. Логічні операції C і оператор розгалуження	35
1.2.7. Перенаправлення файлів введення-виведення	37
1.2.8. Дистрибутив і документація	38
1.2.9. Короткий довідник правил мови C/C++	39
1.3. ПОВТОРЮВАНІ ДІЇ	40
1.3.1. Арифметичні операції, діапазон допустимих значень змінної	42
1.3.2. Оператор безумовного переходу	44
1.3.3. Складений оператор	46
1.3.4. Оператор циклу for	47
1.3.5. Стандартні файли введення-виведення і форматоване виведення мови C	49
1.3.6. Синтаксис	51
1.4. ОБРОБКА ТЕКСТОВИХ ФАЙЛІВ	53
1.4.1. Введення-виведення у мові C	54
1.4.2. Таблиці кодування	55

1.4.3. Неявні перетворення цілих типів	57
1.4.4. Препроцесор і макроси	58
1.4.5. Тестування додатка	59
1.4.6. Подальше покращення програми	60
1.4.7. Оператор циклу while	60
1.5. ОДНОВИМІРНІ МАСИВИ	62
1.5.1. Масиви та їх ініціалізація	62
1.5.2. Операції індексування і sizeof	63
1.5.3. Додаткові арифметичні операції	64
1.5.4. Пріоритет операцій	66
1.5.5. Приклад побічного ефекту інкремента	67
1.6. МАСИВИ І ПРЕПРОЦЕСОР	68
1.6.1. Препроцесор і макроси	68
1.6.2. Незмінювані змінні мови C/C++	70
1.6.3. Кроки побудови консольного додатка	71
1.6.4. Пошук мінімуму і/або максимуму в масиві	72
1.6.5. Висновки	78
1.7. СОРТУВАННЯ	78
1.7.1. Обмін значень двох змінних	78
1.7.2. Макроси з параметрами	80
1.7.3. Сортування вибором	83
1.7.4. Бітові операції	84
Розділ 2. ФУНКЦІЇ ТА РЯДКИ	87
2.1. ФУНКЦІЇ В МОВІ C	87
2.1.1. Визначення функції і оператор return	87
2.1.2. Тип void і операція застосування функції	89
2.1.3. Передача фактичних аргументів у функцію	90
2.1.4. Заголовкові файли	91
2.1.5. Операції і типи	92
2.2. ПЕРЕДАЧА МАСИВІВ У ФУНКЦІЮ, ФУНКЦІЇ У МОВІ C++	93
2.2.1. Передача масивів у функцію	93
2.2.2. Кваліфікатор const	95

2.2.3. Значення формальних параметрів за замовчуванням	95
2.2.4. Перевантаження функцій	96
2.2.5. Вбудовуванні функції	97
2.3. РЕКУРСІЯ	97
2.3.1. Рекурсія як спосіб програмування	97
2.3.2. Повторне включення файлів та умовна компіляція	100
2.3.3. Використання рекурсії для опису даних	101
2.3.4. Оновлений короткий синтаксис мови C	103
2.4. РОБОТА З СИМВОЛАМИ В МОВІ C	105
2.4.1. Невидимі та керуючі (спеціальні) символи	105
2.4.2. Функції із заголовкового файлу ctype.h	107
2.4.3. Вузькі та широкі символи	108
2.4.4. Кирилиця і таблиці кодувань	109
2.5. ПОКАЖЧИКИ ТА ОДНОВИМІРНІ МАСИВИ В МОВІ C	111
2.5.1. Тип даних покажчиків	111
2.5.2. Масиви та покажчики	113
2.5.3. Адресна арифметика	114
2.5.4. Перетворення покажчиків	115
2.5.5. Покажчики на функції	116
2.6. ПОКАЖЧИКИ НА ПОКАЖЧИКИ	118
2.6.1. Масив рядків	118
2.6.2. Аргументи командного рядка	118
2.6.3. Змінні середовища виконання додатка	121
2.6.4. Покажчик на покажчики	121
2.7. РОБОТА З РЯДКАМИ В МОВІ C	124
2.7.1. Рядки мови C	124
2.7.2. Введення-виведення рядків	126
2.7.3. Спільне компілювання файлів у C та C++	126
2.7.4. Функції заголовкових файлів string.h і stdlib.h	129
2.7.5. Синоніми типів	132
2.7.6. Директива препроцесора #pragma	133

2.8. ОПЕРАТИВНА ПАМ'ЯТЬ	134
2.8.1. Класи оперативної пам'яті	134
2.8.2. Захоплення та звільнення оперативної пам'яті в С	136
2.8.3. Захоплення та звільнення оперативної пам'яті в С++	138
2.8.4. Модульне тестування та розробка зверху-вниз	139
2.8.5. Засоби для стеження за витоками пам'яті	144
2.8.5.1. Побудова додатка, специфічного для Microsoft	144
2.8.5.2. Компілятор gcc	147
2.8.6. Новий приклад із передачею матриці у функцію	148

Розділ 3. ОСНОВИ ПРОЦЕДУРНОГО

ПРОГРАМУВАННЯ МОВОЮ С++	150
3.1. АНАЛІЗ ТЕКСТОВИХ ФАЙЛІВ	150
3.1.1. Аргументи командного рядка	150
3.1.2. Функції і макроси для роботи з символами ctype.h	151
3.1.3. Введення-виведення рядків	151
3.1.4. Функція strtok	152
3.1.5. Глобальні змінні, кваліфікатор пам'яті static та інкапсуляція	154
3.2. ПЕРЕРАХУВАННЯ І ОПЕРАТОР SWITCH	155
3.2.1. Перерахування	155
3.2.2. Оператор switch	156
3.2.3. Спільне використання	157
3.2.4. Приклад використання оператора switch в мові С++	157
3.3. ПРЯМИЙ ДОСТУП ДО ФАЙЛІВ	158
3.3.1. Тип FILE і функції для роботи з файлами	158
3.3.2. Двійкове і текстове представлення даних	159
3.3.3. Двійкові й текстові файли	160
3.3.4. Двійковий і текстовий режими доступу до файлів	162

3.4. СТРУКТУРНІ ТИПИ	163
3.4.1. Особливості поняття структури в мовах C і C++	163
3.4.2. Опис синонімів типу typedef	164
3.4.3. Операції на структурах	164
3.4.4. Передача структур у функції і покажчики	165
3.4.5. Вирівнювання структур і операція sizeof	166
3.5. МАСИВИ СТРУКТУР	166
3.5.1. Масиви структур і сортування	166
3.5.2. Об'єднання і ключове слово union	168
3.5.3. Порядок байт у змінних вбудованих типів	168
3.5.4. Об'єднання і поля біт у структурах	170
3.6. ОДНОНАПРЯМЛЕНІ ЗВ'ЯЗАНІ СПИСКИ	171
3.6.1. Структури, які посиляються на себе	171
3.6.2. Стек як однонапрямлений зв'язний список	173
3.6.3. Створення і знищення списків	174

Розділ 4. ДЕЯКІ ОСОБЛИВОСТІ

ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ ...	176
4.1. КЛАСИ, ВІДМІННІСТЬ КЛАСІВ ВІД СТРУКТУР ...	176
4.1.1. Інкапсуляція в класах і структурах	176
4.1.2. Методи і статичні методи класу	177
4.1.3. Конструктор і деструктор	179
4.1.4. Тип посилання	182
4.2. ПЕРЕВАНТАЖЕННЯ ОПЕРАЦІЙ І КЛАСИ	183
4.2.1. Перевантаження конструкторів, конструктор за замовчуванням	183
4.2.1.1. Делегуючий конструктор	185
4.2.2. Перевантаження операції приведення типу	187
4.2.3. Друзі класу	188
4.2.3.1. Повний приклад друзів класу	189
4.2.4. Перевантаження операції введення-виведення ...	190
4.3. НАСЛІДУВАННЯ	192
4.3.1. Наслідування	192
4.3.2. Специфікатори доступу	198
4.3.3. Множинне наслідування	200

4.3.3.1. Віртуальне наслідування	201
4.3.4. Порядок виклику конструкторів і деструкторів ...	202
4.4. НАСЛІДУВАННЯ ТА ПОЛІМОРФІЗМ	202
4.4.1. Поліморфізм	202
4.4.2. Раннє і пізнє зв'язування	202
4.4.3. Перевантаження і перевизначення методів	204
4.4.4. Шаблони функцій	206
4.5. КОНСТРУКТОР ПРИСВОЮВАННЯ І ОПЕРАЦІЯ КОПІЮВАННЯ	207
4.5.1. Ще раз про деструктор	207
4.5.2. Операція присвоювання	208
4.5.3. Конструктор копіювання	209
4.5.4. Перетворення типів у мови C++	211
4.5.4.1. Неявні перетворення	211
4.5.4.2. Явні перетворення	211
Підсумки	214
Список літератури	215
ДОДАТКИ	220
ДОДАТОК А. Приклади програм та програмних засобів	220
A.1. Приклад тексту мовою LaTeX	220
A.2. Кроки розробки програмного забезпечення	220
A.3. Кроки побудови додатка компілятором	221
A.4. Використання компілятора GCC	222
A.5. Використання компілятора MICROSOFT	223
A.6. Використання редактора	223
A.7. Тести для визначення квадранта площини	224
A.8. Р-схема алгоритма для визначення квадранта площини	225
A.9. Скрипт обчислення факторіала мовою PYTHON ...	226
A.10. Скрипт знаходження індексу мінімуму мовою PYTHON	226
A.11. Некоректне використання інкремента	227
A.12. Масиви масивів	227
ДОДАТОК Б. Словник	231

ВСТУП

Підручник присвячений структурному програмуванню та питанням розробки, тестування і документування консольних додатків на компіляторах C++. У першому та другому розділах розглянуто: оператори розгалуження, циклів, безумовного переходу; макроси; функції; перевантаження функцій; функції, що вбудовуються; захоплення пам'яті і системи стеження за витоками пам'яті; обробка текстових файлів та аргументів командного рядка; елементи інкапсуляції, тестування та поліморфізму. У третьому розділі розглянуто прямий доступ до файлів, структури та зв'язні списки. Четвертий розділ присвячен основам об'єктно-орієнтованого програмування*.

З метою узгодження термінології з математикою і традиціями перекладу термінів [37; 45] англійський термін *operator* перекладатимемо як операція, а термін *statement* — як оператор. Тобто “+” — це символ операції, а не оператора.

Як основну літературу, що пов'язана з мовою C, рекомендуємо [37; 45]; з мовою C++ — [36]. Багато корисних прикладів програмування мовою C можна знайти у навчальному посібнику [1]. Матеріал, присвячений мові програмування C++, добре викладено в [49; 44], а також корисним буде з кожного питання звертатися до [33].

* Електронний варіант підручника дивись у [22], а його лабораторний практикум у [19].

Розділ 1

ВВЕДЕННЯ В ПРОГРАМУВАННЯ

1.1. МОВИ ПРОГРАМУВАННЯ І АЛГОРИТМИ

Основоположним для програмування поняттям є інтуїтивне поняття обчислюваності. На сьогодні згідно з тезою А. Черча прийнято вважати, що всі часткові функції, які можна описати за допомогою алгоритму, є частково-рекурсивними функціями Черча [51]. Інакше кажучи, алгоритмом будемо називати те, що можна записати у вигляді частково-рекурсивної функції (те саме може бути записано у вигляді програми для машини Тюрінга [41; 48]). Інтерпретатор для машини Тюрінга можна завантажити за адресою: http://agp1.adr.com.ua/software/_TuringMachine.zip. На даний момент не має значення, що це таке. Важливо лише те, що поняття рекурсивної функції та машини Тюрінга мають точні математичні визначення. І також важливо, що з 30-х років ХХ ст., коли надавалися різні визначення словам обчислення і алгоритму (які виявилися еквівалентними), кожен розглянуту функцію вдавалося представити у вигляді частково-рекурсивної функції Черча. Тому до появи контрприкладу під алгоритмом розумітимемо те, що можна записати за допомогою частково-рекурсивних функцій Черча.

На цьому абстрактні складності закінчуємо і переходимо до простих речей. Не заглиблюючись у строге визначення алгоритму, його створення передбачає наявність таких елементів:

- певна мова;
- деякий письменник, який пише цією мовою;
- деякий виконавець;
- виконавець, який розуміє текст цією мовою і може його виконати за скінченний час;