

Серія «Вища освіта в Україні» заснована у 1999 р.

**Козак Л.І.
Костюк І.В.
Стасевич С.П.**

Основи програмування

НАВЧАЛЬНИЙ ПОСІБНИК

Рекомендовано Міністерством освіти і науки України

«Новий Світ – 2000»

2020

УДК 004.42 (075.8)
ББК 32.973.2
К 59

*Розповсюдження та тиражування
без офіційного дозволу видавництва
заборонено.*

Рекомендовано Міністерством освіти та науки України

Рецензенти:

Мусій Р. С. – доктор фіз.-мат. наук, професор кафедри вищої математики НУ «Львівська політехніка»;

Стадник М. М. – доктор техн. наук, професор, зав. кафедри вищої математики Національного лісотехнічного університету України;

Тумашова О. В. – канд. фіз.-мат. наук, доцент кафедри обчислювальної математики та програмування НУ «Львівська політехніка».

Козак Л. І., Костюк І. В., Стасевич С. П.

К 59 Основи програмування: навчальний посібник – Львів: «Новий Світ-2000», 2020. – 328с.

ISBN978-966-418-159-1

Навчальний посібник містить основні поняття апаратного, програмного забезпечення комп'ютерних систем та операційної системи. Розглядаються основи алгоритмізації, методика розробки алгоритмів, їх застосування, типові алгоритми, показується зв'язок між графічною побудовою алгоритму та його програмним відображенням. Коротко і чітко викладено фундаментальні основи стандарту: інтегроване середовище для роботи, дані, операції, управляючі конструкції, створення функцій, описані цикли, масиви, вказівники, застосування динамічної пам'яті, робота з рядками, файлове введення / виведення. Швидкому засвоєнню принципів програмування сприятимуть різноманітні приклади створення програм, які супроводжуються докладними коментарями, описами алгоритмів та застосованої техніки програмування. Закріпити набуті знання допоможуть вправи. Призначений для студентів та викладачів дисциплін, пов'язаних з програмуванням, а також для тих, хто самостійно вивчає основи програмування.

УДК 004.42(075.8)
ББК 32.973.2
К 59

ISBN978-966-418-159-1

© Козак Л. І., Костюк І. В., Стасевич С. П., 2020

© Новий Світ-2000, 2020

Передмова

На сучасному етапі обчислювальна техніка становить основу науково-технічного прогресу і застосовується в багатьох галузях. Вона є основою розробки засобів автоматизації технологічних процесів, становить перспективну елементну базу сучасних засобів автоматики з програмним управлінням, роботів, АСУ в масштабах підприємств, галузей, регіонів і навіть цілої держави.

Обчислювальна техніка відкрила еру комплексної механізації та автоматизації розумової праці, створення штучного інтелекту. Вже перші ЕОМ, які використовувалися, в основному, як великі арифмометри, вигідно відрізнялися від останніх тим, що не потребували втручання людини в управління обчислювальним процесом та запам'ятовування проміжних результатів. Сучасні ЕОМ вивільняють мозок людини від громіздких обчислень та рутинної роботи, спрямовують його до творчої праці і є незамінним інструментом в роботі будь-якого користувача.

Сьогодні, в епоху майже повної комп'ютеризації, коли основна маса інформації зберігається у безпаперовому вигляді – у пам'яті на електронних носіях інформації, кожний фахівець у будь-якій галузі має бути не тільки ознайомлений з принципами формування масивів електронної інформації, а й уміти її знаходити, користуватися нею, інакше він схожий на людину, яка не вміє ані читати, ані писати.

Для грамотного вирішення сучасних завдань, які виникають на стику наук, щоб відповідати кваліфікаційній характеристиці фахівця, ефективно використовувати комп'ютер і комп'ютерну техніку, сучасний фахівець має не тільки орієнтуватися у методах вирішення таких задач та вміти їх ставити, але й самостійно розробляти алгоритми, знати і вміти використовувати типові алгоритми при їх розв'язуванні, коректно їх описувати зрозумілою комп'ютерною мовою, розумітися на програмах, мати навички розв'язування задач за допомогою комп'ютерної техніки.

Застосування комп'ютерів у різних сферах людської діяльності і залучення до їх використання непрофесіоналів спричинили написання програм мовою, наближеною до людської, і їх переклад на машинні коди самим комп'ютером. Упродовж багатьох років основними характеристиками програми вважали швидкість виконання операцій та її обсяг, оскільки машинна пам'ять тоді коштувала недешево. Але сьогодні комп'ютери зменшилися і подешевшали

настільки, що вартість робочого часу програміста значно перевищує вартість більшості комп'ютерів, які використовують у бізнесі.

Тепер великий попит мають програми, які легко використовуються при зміні умов, пов'язаних із розвитком конкретних завдань, легко перелаштовуються, причому без значних додаткових витрат. Нині таким вимогам відповідає об'єктно-орієнтований підхід (ООП) до програмування, який є потужним засобом вирішення проблеми складності завдань. При ООП програма створюється як сукупність об'єктів, яким притаманні певні властивості та способи поведінки, а не як набір послідовно виконуваних інструкцій.

Серед об'єктно-орієнтованих мов програмування найбільш уживаною є мова C++, яка нині є панівною при розробці програмних продуктів, а об'єктно-орієнтований підхід – пріоритетним. При докладному розгляді синтаксису та семантики значної кількості конструкцій цієї мови не ставилась мета опису всіх її особливостей. Вона використовується як засіб, методи і прийоми, якими має володіти програміст.

Завдання курсу полягає в тому, щоб навчити основ і принципів алгоритмізації та програмування, об'єднуючи логіку побудови алгоритмів з написанням програм, використовуючи мову C++; допомогти засвоїти навички виконання програм в IDE середовищі; навчити створювати функції, працювати із складними типами даних; створювати вказівники, виробляти навички роботи з ними; маніпулювати адресами пам'яті; ефективно застосовувати динамічну пам'ять; організовувати файлове введення/виведення; використовувати маніпулятори при обробці даних у потоці; ознайомити з поширеними типами помилок, які допускають початківці при створенні програм мовою C++, та способами їх усунення; навчити способів обробки винятків.

Вивчення прийомів алгоритмізації та мови програмування C++ у навчальному посібнику супроводжується конкретними прикладами, аналізом програм та коментарями.

Навчальний посібник написано простою, доступною мовою, що дає можливість вивчати програмування «з нуля». У ньому докладно викладено принципи програмування, що стане підґрунтям для створення складніших програм мовою C++.

Посібник призначений для студентів початкових курсів будь-яких спеціальностей, а також тих, хто самостійно вивчає основи програмування об'єктно-орієнтованою мовою C++.

ЗМІСТ

ПЕРЕДМОВА	3
Глава 1. ІНФОРМАЦІЯ ТА КОМП'ЮТЕР	11
Поняття інформації, її властивості. Кількісна оцінка інформації ..	11
Комп'ютер – сучасний пристрій для обробки інформації	12
Запам'ятовування інформації у комп'ютері	13
Системи числення	14
Перетворення чисел з однієї системи числення в іншу	16
Арифметичні операції у двійковій системі числення	21
Одиниці вимірювання інформації у комп'ютері	23
Представлення інформації у комп'ютері	24
Представлення команд	24
Представлення чисел у комп'ютері	25
Представлення символної інформації	28
Покоління комп'ютерів	29
Типи комп'ютерів	32
Апаратне забезпечення комп'ютерної системи обробки інформації	34
Апаратна частина комп'ютера	34
Програмне забезпечення	43
Системне ПЗ	43
Операційна система	43
Завантаження ОС	45
Організація файлової системи	46
Прикладне програмне забезпечення	50
Інструментальне забезпечення	50
Мови програмування як засіб спілкування користувача з комп'ютером	51
Загальна характеристика способів організації програми	54
Характеристики об'єктно-орієнтованих мов	56
Глава 2. ОСНОВИ ПРОГРАМУВАННЯ МОВОЮ C++	59
Розробка програм мовою C++	59
Підготовчий етап програмування	59
Інтегровані середовища розробки програм	60
Налаштування IDE Borland C++	61
Створення файла вихідного коду з розширенням <i>.cpp</i>	62
Компіляція вихідного файла мовою C++ у виконавчий	64

ЗМІСТ

Робота зі створеними вихідними файлами мовою С++	65
Запуск на виконання програми С++	65
Структура програми мовою С++	66
Структура програми	66
Загальне визначення функції та функція <i>main()</i>	67
Оператори	68
Роздільні знаки	69
Об'єкт <i>cout</i> та операція вставки (<<)	69
Керуючі послідовності	70
Директиви препроцесора	71
Коментарі	72
Цикл розробки програми	72
Глава 3. ДАНІ	75
Робота з текстом	75
Загальні відомості	75
Збереження рядків у пам'яті	76
Виведення рядків	77
Маніпулятор <i>endl</i>	77
Змінні та константи	79
Змінні цілого типу	79
Визначення змінної, імена змінних, надання значень змінним ..	81
Використання псевдоніма при визначенні типів	85
Беззнакові типи даних	85
Переповнення беззнакової змінної типу <i>short</i> та знакової змінної типу <i>signed int</i> (знакове ціле)	86
Змінні типу <i>char</i>	88
Перетворення типів	90
Дійсні типи	91
Тип <i>float</i>	91
Тип <i>double</i> та <i>long double</i>	92
Коли і який тип даних слід використовувати	92
Нечислові типи даних	93
Константи	94
Введення даних за допомогою <i>cin</i>	96
Маніпулятор <i>setw</i>	100
Глава 4. ОПЕРАЦІЇ З ДАНИМИ	105
Основні арифметичні операції	105
Операція знаходження залишку	108

Арифметичні операції з присвоєнням	108
Оператори інкремента та декремента	110
Математичні функції	112
Правила виконання операцій у виразах	113
Глава 5. ПОБУДОВА АЛГОРИТМІВ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ	117
Означення алгоритму. Способи відображення алгоритмів	117
Переваги графічного способу відображення алгоритмів	119
Основні етапи алгоритму розв'язку будь-якої задачі. Властивості алгоритму	120
Основні операції обчислювального процесу та їх відповідність геометричним символам	121
Правила та графіка виконання схем алгоритмів	123
Основні алгоритмічні структури	124
Розробка алгоритму лінійної структури та його програмування мовою C++	126
Глава 6. РОЗГАЛУЖЕНИЙ ТА ЦИКЛІЧНИЙ ОБЧИСЛЮВАЛЬНИ ПРОЦЕСИ	135
Визначення розгалуженого обчислювального процесу	135
Прості та складні розгалужені процеси. Відображення умови розгалуження	136
Правила розробки алгоритмів розгалужених структур	137
Приклад розробки алгоритму розгалуженої структури	137
Циклічний обчислювальний процес. Визначення циклічного процесу та методика розробки	138
Типи циклів	141
Тип циклу за кількістю повторень	141
Цикл за кінцевою умовою	143
Цикл за кінцевим індексом	143
Ітераційний цикл	144
Глава 7. ПРОГРАМУВАННЯ АЛГОРИТМІВ РОЗГАЛУЖЕНИХ СТРУКТУР	145
Операції відношення	145
Логічні операції	147
Операція логічного і (&&)	147
Логічне або ()	148
Логічне ні (!)	148
Пріоритети операцій	148

ЗМІСТ

Умовні оператори	149
Загальні відомості	149
Умовний оператор <i>if</i>	149
Оператор <i>if...else</i>	153
Команда умовної операції	155
Оператор <i>switch</i>	156
Оператор <i>break</i>	158
<i>Switch</i> і символічні змінні	158
Програмування алгоритмів циклічної структури	159
Загальні відомості	159
Цикл з лічильником <i>for</i>	160
Варіації на тему циклу <i>for</i>	162
Модифікація циклу <i>for</i> для використання змінних та констант ...	165
Цикл <i>while</i>	167
Цикл <i>do-while</i>	169
Глава 8. ТИПОВІ АЛГОРИТМИ	171
Обчислення значень функції	171
Обчислення суми та добутку	181
Алгоритм упорядкування трьох чисел	186
Обчислення суми членів нескінченного ряду із заданою точністю	188
Знаходження найбільшого та найменшого значень	189
Знаходження найбільшого та найменшого у масиві	191
Глава 9. ТИПОВІ АЛГОРИТМИ ПРИ РОБОТІ З МАСИВАМИ ..	195
Загальне поняття масиву	195
Алгоритм введення елементів масиву у пам'ять комп'ютера ...	196
Алгоритм накопичення суми, добутку елементів одномірного масиву	200
Приклади деяких алгоритмів	202
Створення випадкових чисел	213
Глава 10. ФУНКЦІЇ	215
Створення функцій	215
Принцип модульності програм та роль функції <i>main()</i> у них	215
Основні поняття та елементи функції	216
Бібліотечні функції та функції користувача	217
Оголошення функції	218
Визначення функції	220
Виклик функції	222

Створення функції	223
Вбудовані функції	225
Параметри та аргументи функції	227
Передача аргументів у функцію	227
Передача аргументів у функцію за значенням	232
Передача аргументів за посиланням	235
Перевантаження функцій	236
Рекурсія	238
Шаблони функції	241
Макровизначення	243
Глава 11. ПАМ'ЯТЬ: ВИЗНАЧЕННЯ АДРЕСИ, ВКАЗІВНИКИ, СТЕК І ДИНАМІЧНА ПАМ'ЯТЬ	245
Операція визначення адреси	245
Поняття вказівника. Оголошення, іменування та ініціалізація вказівника	248
Оператор непрямого доступу до значення змінної	250
Вказівники і масиви	254
Арифметичні операції з вказівниками	255
Стек і динамічна пам'ять	257
Оголошення змінних у вільній пам'яті	259
Реалізація функції <i>Obmin()</i> із використанням вказівників	265
Створення масиву у вільній пам'яті	267
Глава 12. ВВЕДЕННЯ, ВИВЕДЕННЯ ТА ФАЙЛИ	269
Оглядовий аналіз введення-виведення у C++	269
Інкапсуляція	270
Буферизація	270
Класи	271
Стандартні об'єкти введення-виведення	271
Об'єкт <i>cin</i>	271
Методи об'єкта <i>cin</i>	272
Функції – члени <i>peek()</i> та <i>putback()</i>	273
Об'єкт <i>cout</i>	274
Маніпулятори, прапорці	274
Функція <i>cout.width()</i> та встановлення символів заповнення ..	275
Прапорці форматування та маніпулятори	276
Введення та виведення у файл	279
Об'єкт <i>ofstream</i>	279
Зчитування даних з файла	280

ЗМІСТ

Виведення даних у файл	281
Зміна поведінки об'єкта <i>ofstream</i>	283
Глава 13. РЯДКИ. СПОСОБИ ЗБЕРІГАННЯ РЯДКА	287
Ініціалізація масиву рядками	288
Проблеми читання рядка з клавіатури	291
Рядково-орієнтоване введення, <i>getline()</i> та <i>get()</i>	291
Поєднання рядкового та числового введення	294
Рядки як клас <i>STRING</i>	295
Глава 14. ПОМИЛКИ ТА ОБРОБКА ПОМИЛОК	305
I. Типи помилок	305
II. Помилки компіляції	305
III. Помилки часу виконання та логічні помилки	312
IV. Використання середовища	321
ЛІТЕРАТУРА	324