

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЖИТОМИРСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ФРАНКА

**С. С. ЖУКОВСЬКИЙ, Т. А. ВАКАЛЮК**

# **Об'єктно-орієнтоване програмування мовою С++**

*навчально-методичний посібник  
для студентів напрямку 6.040302 Інформатика\**

Житомир 2016

УДК 004.42+004.432.2(076.5)

ББК 73р

Ж86

*Затверджено Вченою радою Житомирського державного університету імені Івана Франка протокол № 4 від 01.11.2016 р.*

### **Рецензенти:**

**Шевчук Л.Д.** – кандидат педагогічних наук, доцент, заступник завідувача кафедри математики, інформатики та методики навчання ДВНЗ «Переяслав-Хмельницький державний педагогічний університет імені Григорія Сковороди»;

**Медведєва М.О.** – кандидат педагогічних наук, доцент, доцент кафедри вищої математики та методики навчання математики Уманського державного педагогічного університету імені Павла Тичини;

**Мінтій І.С.** – кандидат педагогічних наук, доцент кафедри інформатики та прикладної математики Криворізького державного педагогічного університету.

**Ж86 Жуковський С.С., Вакалюк Т.А.**

Об'єктно-орієнтоване програмування мовою С++. Навчально-методичний посібник для студентів напряму 6.040302 Інформатика\*.  
– Житомир: Вид-во ЖДУ, 2016. – 100 с.

Посібник призначений для використання студентами під керівництвом викладача на лекціях, практичних та лабораторних заняттях. Посібник містить лекційний курс та лабораторний практикум із об'єктноорієнтованого програмування мовою С++. Викладений матеріал відповідає діючій програмі з програмування для спеціальності «Інформатика»

Для студентів фізико-математичних спеціальностей вищих педагогічних закладів, вчителів інформатики загальноосвітніх шкіл.

УДК 004.42+004.432.2(076.5)

ББК 73р

© Жуковський С.С., Вакалюк Т.А., 2016

## ЗМІСТ

<b>ТЕОРЕТИЧНІ ОСНОВИ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ МОВОЮ C++ .....</b>	<b>4</b>
Структури .....	4
Робота з текстовими та двійковими файлами .....	4
Використання списків .....	5
Перевантаження операцій .....	6
Успадкування класів .....	6
Виключні ситуації .....	7
Клас String .....	7
Контейнери.....	8
<b>ЛАБОРАТОРНИЙ ПРАКТИКУМ .....</b>	<b>9</b>
Лабораторна робота №1. Розробка програм з використанням структур....	9
Лабораторна робота №2. Розробка програм з використанням текстових та двійкових файлів .....	10
Лабораторна робота №3. Розробка програм з використанням списків ....	11
Лабораторна робота №4. Розробка програм з використанням класів .....	12
Лабораторна робота №5. Перевантаження операцій.....	13
Лабораторна робота №6. Множинне успадкування мовою C++.....	13
Лабораторна робота №7. Виключні ситуації.....	14
Лабораторна робота №8. Клас String.....	15
Лабораторна робота №9. Контейнери .....	16
<b>ТЕСТОВІ ЗАВДАННЯ .....</b>	<b>17</b>
<b>РЕКОМЕНДОВАНА ЛІТЕРАТУРА .....</b>	<b>18</b>

## ТЕОРЕТИЧНІ ОСНОВИ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ МОВОЮ С++

---

### Структури

**Мета:** ознайомити студентів з основними поняттями структур.

**Професійна спрямованість:** дана лекція є складовою частиною професійної підготовки вчителя інформатики до майбутньої професійної діяльності.

**Наочність:** схематичні зображення.

**Запитання для самоаналізу та самоперевірки:**

1. Який синтаксис опису структури?
2. Як розподіляється пам'ять у структурі?
3. Як можна забезпечити доступ до елемента структури?
4. Як можна забезпечити доступ до елемента структури через вказівник?
5. Як описати змінну структурованого типу?
6. Як описати в структурі змінну структурованого типу?
7. Що таке вкладені структури?
8. Як визначити розмір структури?
9. Що таке покажчики на структуру?
10. Що таке масив структур?
11. Що таке бітові поля?
12. Що таке об'єднання?

**Текст лекції.**

### Робота з текстовими та двійковими файлами

**Мета:** ознайомити студентів з основними поняттями розробки та описання програм з використанням текстових та двійкових файлів.

**Професійна спрямованість:** дана лекція є складовою частиною професійної підготовки вчителя інформатики до майбутньої професійної діяльності.

**Наочність:** схематичні зображення.

**Запитання для самоаналізу та самоперевірки:**

1. Що таке текстовий (бінарний) файл?
2. Як оголосити файлову змінну (stdio.h, fstream)?
3. Як відкрити файл для читання, запису, дозапису (stdio.h, fstream)?
4. Як записати дані у текстовий файл (stdio.h, fstream)?
5. Як зчитати дані з текстового файлу (stdio.h, fstream)?
6. Як записати дані у двійковий файл ?
7. Як зчитати дані з двійкового файлу ?
8. Як встановити позицію запису, зчитування даних з файлу (у файл)?
9. Як визначити позицію зчитування(запису)?
10. Як закрити текстовий, (бінарний) файл?
11. Як визначити кінець файлу?

*Текст лекції.*

## Використання списків

**Мета:** Набути умінь та навички розробки та описання програм з використанням списків, текстових та двійкових файлів.

**Професійна спрямованість:** дана лекція є складовою частиною професійної підготовки вчителя інформатики до майбутньої професійної діяльності.

**Наочність:** схематичні зображення.

**Запитання для самоаналізу та самоперевірки:**

1. Який синтаксис опису структури?
1. Як розподіляється пам'ять у структурі?
2. Як можна забезпечити доступ до елемента структури?
3. Що таке список?
4. Як реалізувати список мовою Cі?
5. Як додати елемент в кінець списку, в середину списку, на початок списку?
6. Що таке стек?
7. Як реалізувати стек мовою Cі?
8. Що таке черга?
9. Як реалізувати чергу мовою Cі?
10. Яка перевага списку над масивом?
11. Що таке файл?
12. З якими файлами можна працювати в C++?
13. Як відкрити текстові файли для читання, запису?
14. Як відкрити двійкові (бінарні) файли для читання, запису?
15. Які переваги двійкових файлів перед текстовими і навпаки?
16. Як закрити файл?
17. Як звернутись до певного елемента бінарного файлу?
18. Як перейти на початок бінарного файлу?
19. Як перейти на кінець бінарного файлу?

*Текст лекції.*

## Поняття класу, конструктор , деструктор

**Мета:** Набути умінь та навички розробки та описання програм з використанням класів, методів класу, конструкторів, деструкторів.

**Професійна спрямованість:** дана лекція є складовою частиною професійної підготовки вчителя інформатики до майбутньої професійної діяльності.

**Наочність:** схематичні зображення.

**Запитання для самоаналізу та самоперевірки:**

1. Що таке клас в С++?
2. Який синтаксис опису класу?
3. Що таке поля, методи класу?
4. Які є специфікатори доступу класу, їхнє призначення?
5. Як можна забезпечити доступ до елементів класу?
6. Що таке конструктор? Які правила створення та роботи конструктора ?
7. Що таке конструктор позамовчунні, конструктор копіювання, конструктор перетворення?
8. Що таке деструктор? Які правила створення та роботи деструктора ?
9. Які існують ініціалізації елементів у конструкторах?
10. Який порядок виклику конструкторів та деструкторів?

## Перевантаження операцій

**Мета:** Набути умінь та навички розробки та описання програм з використанням перевантаження операцій.

**Професійна спрямованість:** дана лекція є складовою частиною професійної підготовки вчителя інформатики до майбутньої професійної діяльності.

**Наочність:** схематичні зображення.

**Запитання для самоаналізу та самоперевірки:**

1. Що таке перевантаження операцій?
2. Який синтаксис перевантаження?
3. Які операції неможна перевантажувати?
4. За допомогою якої операції виконується перевантаження операцій?
5. Коли необхідно перевантажувати операцію присвоєння?
6. Як передаються параметри під час перевантаження операції?

**Текст лекції.**

## Успадкування класів

**Мета:** отримання практичних навичок при використанні множинного успадкування мовою С++.

**Професійна спрямованість:** дана лекція є складовою частиною професійної підготовки вчителя інформатики до майбутньої професійної діяльності.

**Наочність:** схематичні зображення.

**Запитання для самоаналізу та самоперевірки:**

1. Що таке успадкування класів?
2. Який синтаксис успадкування класів?
3. Принцип керування доступом елементів класу при успадкуванні?
4. Як успадковуються конструктори, деструктори.
5. Як викликаються конструктори, деструктори успадкованих класів?

**Рекомендована література:**

1. Т.А.Павловська С/С++ програмування на языке високого уровня.(ст..200-200)
2. Харви Дейтел, Пол\_Дейтел Как программировать на С++ (ст..551-596)

## Виключні ситуації

**Мета:** Набути уміння та навички розробки та описання програм з використанням обробки виняткових(виключних) ситуацій.

**Професійна спрямованість:** дана лекція є складовою частиною професійної підготовки вчителя інформатики до майбутньої професійної діяльності.

**Наочність:** схематичні зображення.

**Запитання для самоаналізу та самоперевірки:**

1. Що таке виключні ситуації.
2. Синтаксис виключних ситуацій.
3. Як контролюються виключні ситуації
4. Як відбувається перехват виключних ситуацій

**Текст лекції.**

## Клас String

**Мета:** Набути уміння та навички розробки та описання програм з використанням класу String..

**Професійна спрямованість:** дана лекція є складовою частиною професійної підготовки вчителя інформатики до майбутньої професійної діяльності.

**Наочність:** схематичні зображення.

**Запитання для самоаналізу та самоперевірки:**

1. Що таке клас string?

2. Які конструктори існують в класі string?
3. Які операції допустимі для об'єктів класу string?
4. Як реалізуються методи обробки рядка, об'єкту string?
5. Які методи обробки рядка існують?
6. Які методи переводять з типу string в тип char?

*Текст лекції.*

## Контейнери

**Мета:** Набути умінь та навички розробки та описання програм з використанням контейнерів та бібліотеки <algorithm>

**Професійна спрямованість:** дана лекція є складовою частиною професійної підготовки вчителя інформатики до майбутньої професійної діяльності.

**Наочність:** схематичні зображення.

**Запитання для самоаналізу та самоперевірки:**

1. Що таке контейнер.
2. Які є контейнери. Їх призначення.
3. Для чого призначений вектор? Які методи він підтримує?
4. Для чого призначений черга? Які методи він підтримує?
5. Для чого призначений список? Які методи він підтримує?
6. В якому з контейнерів можна додати, видалити елементи в кінець.
7. В якому з контейнерів можна додати, видалити на початок?
8. В якому з контейнерів можна додати, вставити елементи всередину.

*Текст лекції.*



## ЛАБОРАТОРНИЙ ПРАКТИКУМ

---

### Лабораторна робота №1. Розробка програм з використанням структур

**Мета:** Набути уміння та навички розробки та описання програм обробки з використанням структур.

**Програмне забезпечення:** Середовище програмування DEV C++, Visual C++.

#### Контрольні питання.

1. Який синтаксис опису структури?
2. Як розподіляється пам'ять у структурі?
3. Як можна забезпечити доступ до елемента структури?
4. Як можна забезпечити доступ до елемента структури через вказівник?
5. Як описати змінну структурованого типу?
6. Як описати в структурі змінну структурованого типу?
7. Що таке вкладені структури?
8. Як визначити розмір структури?
9. Що таке покажчики на структуру?
10. Що таке масив структур?
11. Що таке бітові поля?
12. Що таке об'єднання?

#### Практична частина.

## **Лабораторна робота №2. Розробка програм з використанням текстових та двійкових файлів**

*Мета:* Набути уміння та навички розробки та описання програм з використанням текстових та двійкових файлів.

*Програмне забезпечення:* Середовище програмування DEV С++, Visual С++ .

### **Контрольні питання.**

1. Що таке текстовий (бінарний) файл?
2. Як оголосити файлову змінну (stdio.h, fstream)?
3. Як відкрити файл для читання, запису, дозапису (stdio.h, fstream)?
4. Як записати дані у текстовий файл (stdio.h, fstream)?
5. Як зчитати дані з текстового файлу (stdio.h, fstream)?
6. Як записати дані у двійковий файл ?
7. Як зчитати дані з двійкового файлу ?
8. Як істановити позицію запису, зчитування даних з файлу (у файл)?
9. Як визначити позицію зчитування(запису)?
10. Як закрити текстовий, (бінарний) файл?
11. Як визначити кінець файлу?

### ***Завдання до лабораторної роботи.***

## Лабораторна робота №3. Розробка програм з використанням списків

**Мета:** Набути уміння та навички розробки та описання програм з використанням списків, текстових та двійкових файлів.

**Програмне забезпечення:** DEV C++, Visual C++.

### Контрольні питання.

1. Який синтаксис опису структури?
2. Як розподіляється пам'ять у структурі?
3. Як можна забезпечити доступ до елемента структури?
4. Що таке список?
5. Як реалізувати список мовою Cі?
6. Як додати елемент в кінець списку, в середину списку, на початок списку?
7. Що таке стек?
8. Як реалізувати стек мовою Cі?
9. Що таке черга?
10. Як реалізувати чергу мовою Cі?
11. Яка перевага списку над масивом?
12. Що таке файл?
13. З якими файлами можна працювати в C++?
14. Як відкрити текстові файли для читання, запису?
15. Як відкрити двійкові (бінарні) файли для читання, запису?
16. Які переваги двійкових файлів перед текстовими і навпаки?
17. Як закрити файл?
18. Як звернутись до певного елемента бінарного файлу?
19. Як перейти на початок бінарного файлу?
20. Як перейти на кінець бінарного файлу?

### Зміст завдання

## **Лабораторна робота №4. Розробка програм з використанням класів**

*Мета:* Набути уміння та навички розробки та описання програм з класами. Написання конструкторів, деструкторів, методів(функцій-членів)

*Програмне забезпечення:* Dev C++, Visual C++

### **Контрольні питання.**

1. Який синтаксис опису класу?
2. Що таке поля, методи класу?
3. Які є специфікатори доступу класу, їхнє призначення?
4. Як можна забезпечити доступ до елементів класу?
5. Що таке конструктор? Які правила створення та роботи конструктора ?
6. Що таке конструктор позамовчунні, конструктор копіювання?
7. Що таке деструктор? Які правила створення та роботи деструктора ?
8. Які існують ініціалізації елементів у конструкторах?
9. Який порядок виклику конструкторів та деструкторів?

### ***Зміст завдання***

## Лабораторна робота №5. Перевантаження операцій

*Мета:* Набути уміння та навички розробки та описання програм з використанням перевантаження операцій.

*Програмне забезпечення:* Dev C++, Visual C++

### Теоретичні питання.

1. Що таке перевантаження операцій?
2. Який синтаксис перевантаження?
3. Які операції неможна перевантажувати?
4. За допомогою якої операції виконується перевантаження операцій?
5. Коли необхідно перевантажувати операцію присвоєння?
6. Як передаються параметри під час перевантаження операції?

### Зміст завдання

## Лабораторна робота №6. Множинне успадкування мовою C++.

*Мета:* отримання практичних навиків при використанні множинного успадкування мовою C++

*Програмне забезпечення:* Dev C++, Visual C++

### Теоретичні питання.

1. Що таке успадкування класів?
2. Який синтаксис успадкування класів?
3. Принцип керування доступом елементів класу при успадкуванні?
4. Як успадковуються конструктори, деструктори.
5. Як викликаються конструктори, деструктори успадкованих класів?

### Зміст завдання

## **Лабораторна робота №7. Виключні ситуації**

*Мета:* Набути умінь та навички розробки та описання програм з використанням обробки виняткових(виключних) ситуацій

*Програмне забезпечення:* Dev C++, Visual C++

### **Теоретичні питання.**

1. Що таке виключні ситуації.
2. Синтаксис виключних ситуацій.
3. Як контролюються виключні ситуації
4. Як відбувається перехват виключних ситуацій

### *Зміст завдання*

## Лабораторна робота №8. Клас String

*Мета:* Набути умінь та навички розробки та описання програм з використанням класу String..

*Програмне забезпечення:* Dev C++, Visual C++

### **Теоретичні питання.**

1. Що таке клас string?
2. Які конструктори існують в класі string?
3. Які операції допустимі для об'єктів класу string?
4. Як реалізуються методи обробки рядка, об'єкту string?
5. Які методи обробки рядка існують?
6. Які методи переводять з типу string в тип char?

### *Зміст завдання*

## Лабораторна робота №9. Контейнери

*Мета:* Набути уміння та навички розробки та описання програм з використанням контейнерів та бібліотеки <algorithm>.

*Програмне забезпечення:* Dev C++, Visual C++

### **Теоретичні питання.**

1. Що таке контейнер.
2. Які є контейнери. Їх призначення.
3. Для чого призначений вектор? Які методи він підтримує?
4. Для чого призначений черга? Які методи він підтримує?
5. Для чого призначений список? Які методи він підтримує?
6. В якому з контейнерів можна додати, видалити елементи в кінець.
7. В якому з контейнерів можна додати, видалити на початок?
8. В якому з контейнерів можна додати, вставити елементи всередину.

### *Зміст завдання*



## ТЕСТОВІ ЗАВДАННЯ

---

1. Конструктор класу призначений для:

- а) Створення та ініціалізації об'єкта.
- б) Видалення об'єкта
- в) Модифікації об'єкта
- г) Опису об'єкта

2. Які операції неможна перевантажувати?

- а) >
- б) [ ]
- в) ::
- г) ++

3. Як оголосити файлову змінну використовуючи бібліотеку `stdio.h`?

- а) `file f;`
- б) `FILE f;`
- в) `TEXT f;`
- г) `ifstream f;`

4. Як правильно відкрити текстовий файл для читання, використовуючи можливості бібліотеки `stdio.h`?

- а) `f=fopen("text.txt", "rb");`
- б) `FILE f;`
- в) `f=fopen("text.dat", "r");`
- г) `ifstream f ("text.txt");`

5. Для встановлення вказівника в кінець бінарного файлу використовується функція:

- а) `fseek(f, 0, SEEK_END);`
- б) `ftell(f);`
- в) `while(!feof(f){.....}`
- г) `fseek(f, 0, SEEK_CUR);`

6. Для визначення поточної позиції вказівника у файлі використовується функція:

- а) `fseek(f, 0, SEEK_END);`
- б) `ftell(f);`
- в) `while(!feof(f){.....}`
- г) `fseek(f, 0, SEEK_CUR);`

## РЕКОМЕНДОВАНА ЛІТЕРАТУРА

---

1. Керниган Б, Ридчи Д. Язык программирования Си. М. Финансы и статистика.1988.
2. Павловская Т.А. С/С++ Программирование на языке высокого уровня: учебное пособие для вузов СПб, Питер, 2002;
3. Павловская Т.А. С/С++ Структурное программирование. Практикум. учебное пособие для вузов СПб, Питер, 2005;
4. Подбельский В.В. Язык Си++. Финансы и статистика.2001.
5. Войтенко В.В. Морозов А.В. С/С++ теорія та практика ЖДТУ 2003
6. Глинський Я.М., Анохін В.Є., Ряжська В.А. С++ і С++ Builder. Навчальний посібник. – Львів; Деол, СПД Глинський, 2003. – 192 с.
7. Пол Лукас С++ под рукой. КиевНИПФ «Диа Софт» 1993
8. [www.e-olimp.com](http://www.e-olimp.com) – Інтернет-портал організаційно-методичного забезпечення дистанційних олімпіад з програмування.
9. Кнут Д. Искусство программирования для ЭВМ. -М.: Мир, 1976. -Т.2.

Об'єктно-орієнтоване програмування мовою C++

## **ДЛЯ НОТАТОК**

Об'єктно-орієнтоване програмування мовою C++

## **ДЛЯ НОТАТОК**

Об'єктно-орієнтоване програмування мовою С++

Навчально-методичне видання

**ЖУКОВСЬКИЙ Сергій Станіславович,  
ВАКАЛЮК Тетяна Анатоліївна**

**ОБ'ЄКТНО-ОРІЄНТОВАНЕ  
ПРОГРАМУВАННЯ МОВОЮ С++**

*Навчально-методичний посібник для студентів  
напряму 6.040302 Інформатика\**

Надруковано з оригінал-макета автора

Підписано до друку 03.11.16. Формат 60x90/16. Папір офсетний.

Гарнітура Times New Roman. Друк різнографічний.

Ум. друк. арк. 5,8. Обл. вид. арк. 3,5. Наклад 300. Зам. 83.

---

Видавець і виготовлювач

Видавництво Житомирського державного університету імені Івана Франка

м. Житомир, вул. Велика Бердичівська, 40

Свідоцтво суб'єкта видавничої справи:

серія ЖТ №10 від 07.12.04 р.

електронна пошта (E-mail): zu@zu.edu.ua