

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЦЕНТРАЛЬНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ  
УНІВЕРСИТЕТ

ФАКУЛЬТЕТ АВТОМАТИКИ ТА ЕНЕРГЕТИКИ

КАФЕДРА ПРОГРАМУВАННЯ КОМП'ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ

## **Кросплатформені мови програмування**

Методичні вказівки до виконання самостійних робіт

з елементами кредитно - модульної  
системи організації навчального процесу

*для студентів денної форми навчання  
за спеціальністю 123 «Комп'ютерна інженерія»*

Укладачі:

Доцент

Доцент

Смірнов В.В.

Смірнова Н.В.

Кропивницький  
2022 рік

Кросплатформені мови програмування : методичні вказівки до виконання самостійних робіт для студентів денної форми за спеціальністю 123 «Комп'ютерна інженерія» / уклад. : В.В. Смірнов, Н.В. Смірнова ; М-во освіти і науки України, Центральнoукраїн. нац. техн. ун-т. - Кропивницький : ЦНТУ, 2022. – 39 с.

***Затверджено на засіданні кафедри програмування комп'ютерних систем та мереж: 10 травня 2022 р., протокол № 8***

Укладачі:

**Смірнова Наталя Володимирівна**, к.т.н., доцент кафедри кібербезпеки та програмного забезпечення

**Смірнов Володимир Вікторович**, к.т.н., доцент кафедри кібербезпеки та програмного забезпечення

Для студентів денної форми навчання, що вивчають навчальну дисципліну “Кросплатформені мови програмування” за спеціальністю 123 «Комп'ютерна інженерія».

У стислій формі викладені основні принципи побудови об'єктно-орієнтованих подієво-керованих додатків на мові програмування Java в середовищі розробки NET\_Beans IDE.

Представлені практичні рішення навчальних завдань для кращого освоєння досліджуваного матеріалу, представлені варіанти навчальних завдань для самостійного придбання практичних навичок.

*Примітка: Дані методичні вказівки є інтелектуальною власністю авторів. Методичні вказівки можуть використовуватися у навчальному процесі ЦНТУ, копіюватися, розмножуватися і поширюватися без обмежень.*

*Будь-яке внесення змін і доповнень до тексту методичних вказівок без письмового дозволу авторів на підставі законів України "Про інтелектуальну власність" і "Про авторське право і суміжні права" кваліфікується як порушення авторських прав.*

© В.В. Смірнов, Н.В. Смірнова / 2022 рік

© ЦНТУ, кафедра програмування комп'ютерних систем та мереж

## Вступ

У сучасний період затребувані високий рівень знань, академічна і соціальна мобільність, професіоналізм фахівців, готовність до самоосвіти та самовдосконаленню. У зв'язку із цим повинні змінитися підходи до планування, організації навчально – виховної роботи, у тому числі і самостійної роботи студентів. Насамперед, це стосується зміни характеру та змісту навчального процесу, переносу акценту на самостійний вид діяльності, який є не просто самоціллю, а засобом досягнення глибоких і міцних знань, інструментом формування в студентів активності та самостійності.

Метою методичних рекомендацій є підвищення ефективності навчального процесу, у тому числі завдяки самостійній роботі, у якій студент стає активним суб'єктом навчання, що означає:

- здатність займати у навчанні активну позицію;
- готовність мобілізувати інтелектуальні і вольові зусилля для досягнення навчальних цілей;
- уміння проектувати, планувати і прогнозувати навчальну діяльність;
- звичку ініціювати свою пізнавальну діяльність на основі внутрішньої позитивної мотивації;
- усвідомлення своїх потенційних навчальних можливостей і психологічну готовність скласти програму дій по саморозвитку.

## **МЕТА ТА ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

Основна мета курсу полягає в придбанні досконалих знань і навичок роботи в середовищі Кросплатформеної мови програмування Java з застосуванням сучасних технологій та інструментальних засобів.

Внаслідок проведення лекцій студенти повинні отримати теоретичні знання та методику ефективної роботи з сучасними методами створення об'єктно-орієнтованих керованих подіями додатків

### **Завдання вивчення дисципліни**

- Вивчення теоретичних основ об'єктно-орієнтованого проектування;
- Вивчення теоретичних основ об'єктно-орієнтованого програмування;
- Вивчення теоретичних основ методів створення багатокomпонентних додатків;
- Вирішення завдань роботи з файлами та дисками;
- Вирішення завдань створення елементів управління програми з використанням бібліотеки Swing;
- Набуття практичних навиків в сфері програмування багатопотокових додатків на основі технології Java.

**Предметом навчальної дисципліни** є створення багатопотокових об'єктно-орієнтованих подієво-керованих додатків в середовищі програмування Java.

## **У результаті вивчення навчальної дисципліни студент повинен**

### **знати:**

1. Основи міжплатформеної технології Java.
2. Принципи створення подієво-керованих програм.
3. Основи функціонування багатопотокових додатків.

### **вміти:**

1. Вирішувати завдання створення подієво-керованих програм.
2. Вирішувати завдання створення об'єктно-орієнтованих додатків в середовищі розробки JavaBeans.
3. Вирішувати завдання управління потоками в процесі виконання багатопотокового додатку.
4. Програмувати ДОДАТКИ використанням технології JavaBeans і бібліотеки Swing.

## **Самостійна робота**

Для опанування матеріалу дисципліни «Кросплатформені мови програмування» окрім лекційних та лабораторних занять, тобто аудиторного навантаження, значна увага приділяється самостійній роботі.

До основних видів самостійної роботи студента відносимо:

1. Вивчення лекційного матеріалу.
2. Робота з літературними джерелами.
3. Розв'язання практичних задач за індивідуальними варіантами.
4. Підготовка до модульних, підсумкового контролю, екзамену

## РОЗПОДІЛ НАВЧАЛЬНОГО ЧАСУ САМОСТІЙНОЇ РОБОТИ СТУДЕНТА

№ з/п	Назва теми	Кількість годин
1	Освоєння IDE JavaBeans	7
2	Створення класів Java	7
3	Внутрішні класи і конструктори	7
4	Бібліотека Swing - створення додатків і обробка подій	6
5	Бібліотека Swing - створення GUI	6
6	Обробка виняткових ситуацій	6
7	Робота з файловою системою	6
8	Багатопотокове програмування	6
9	Бібліотечні функції Java	6
10	класи утиліт	7
11	Графічні засоби Java	7
12	Друк документів з Java	7
<b>Всього</b>		<b>78</b>

## ТЕМИ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

### Змістовий модуль 1

Введення в Java. Класи. Вкладені і внутрішні класи. Бібліотека Swing. Виключення.

#### Тема 1.1. Введення в Java

- Введення в Java
- Типи даних
- Керуючі конструкції
- Створення проекту
- Консольний додаток
- Компіляція і запуск програми

#### Тема 1.2. Класи

- Основи класів
- Об'ява об'єктів
- Конструктори
- Члени класу
- Повернення значення
- Передача параметрів
- Метод finalize ( )
- Перевантаження методів
- Перевантаження конструкторів
- Використання об'єктів як параметри
- Повернення об'єктів
- Ключове слово static
- Ключове слово final

### **Тема 1.3. Вкладені і внутрішні класи**

- Наслідування
- Ключове слово `super`
- Порядок виклику конструкторів
- Пакети та інтерфейси
- Модифікатори доступу `private`, `public` і `protected`
- Імпорт пакетів
- Визначення інтерфейсу
- Реалізація інтерфейсів
- Змінні в інтерфейсах

### **Тема 1.4. Бібліотека Swing**

- Архітектура MVC
- Додаток Swing
- Обробка подій
- Класи `JLabel` і `ImageIcon`
- Клас `JTextField`
- Клас `JButton`
- Клас `JCheckBox`
- Клас `JRadioButton`
- Клас `JTabbedPane`
- Клас `JScrollPane`
- Клас `JList`
- Клас `JComboBox`
- Клас `JTable`
- Допоміжні класи



## **Тема 1.5. Виключення**

- Обробка виняткових ситуацій
- Блоки перехоплення виключення
- Оператор throw
- Ієрархія класів - виключень
- Створення власних виключень

## **Змістовий модуль 2**

Файли та друк. Багатопотокове програмування. Бібліотечні функції Java. Графіка 2D

### **Тема 2.1. Файли та друк**

- Друк документів
- Потоки введення / виведення
- Консольне введення / виведення
- Клас Console
- Форматоване виведення
- Файлове введення / виведення
- Специфікації виведення
- Отримання властивостей файлу
- Робота з файлом засобами NIO2
- Серіалізація об'єктів.
- Друк в Java
- Друк засобами Java 2D

### **Тема 2.1. Багатопотокове програмування**

- Пріоритети потоків
- Клас Thread і інтерфейс Runnable
- Реалізація інтерфейсу Runnable
- Розширення класу Thread

## **Тема 2.2. Бібліотечні функції Java**

- Пакет java.util
- Клас Runtime
- Клас Calendar, клас Time
- Клас Timer
- Класи утиліт

## **Тема 2.3. Графіка 2D**

- Графічні засоби Java
- Малювання фігур
- Управління виведенням мультимедіа

## ТЕМИ РЕФЕРАТІВ

№ п/п	Теми рефератів
	<b>Змістовий модуль 1</b>
1.	<b>Індивідуальне завдання 1:</b> <b>Основи Java</b> 1. Дати короткий огляд основних типів даних Java. 2. Дати опис середовища розробки програм JavaBeans. 3. Дати короткі рекомендації зі створення проекту, компіляції та запуску додатка. 4. Висновки.
2.	<b>Індивідуальне завдання 2:</b> <b>Вкладені і внутрішні класи</b> 1. Дати короткий огляд класів и об'єктів. 2. Дати опис принципів роботи параметризованих конструкторів і конструкторів за замовчуванням. 3. Дати короткі рекомендації по використанню ключових слів static и final. 4. Висновки.
3.	<b>Індивідуальне завдання 3:</b> <b>Архітектура MVC.</b> 1. Дати короткий огляд архітектури MVC. 2. Дати опис принципів створення додатків Swing. 3. Дати короткі рекомендації по застосуванню бібліотеки Swing. 4. Висновки.
4.	<b>Індивідуальне завдання 4:</b> <b>Бібліотека Swing</b> 1. Дати короткий огляд типів і класів подій в Java. 2. Дати опис принципів роботи системи обробки подій Java. 3. Дати короткі рекомендації з використання класів JLabel и ImageIcon 4. Висновки.

№ п/п	Теми рефератів
5.	<b>Індивідуальне завдання 5:</b> <b>Бібліотека Swing</b> 1. Дати короткий огляд класов с JtextField и JButton. 2. Дати опис принципів роботи класів JCheckBox та JList. 3. Дати короткі рекомендації з використання класів JtabbedPane та JComboBox. 4. Висновки.
	<b>Змістовий модуль 2</b>
6.	<b>Індивідуальне завдання 6:</b> <b>Бібліотека Swing</b> 1. Дати короткий огляд класа JradioButton. 2. Дати опис принципів роботи класа JTable. 3. Дати короткі рекомендації з використання класу Runtime 4. Висновки.
7.	<b>Індивідуальне завдання 7:</b> <b>Виключення</b> 1. Дати короткий огляд технології обробки виняткових ситуацій у Java. 2. Дати опис принципів роботи блоків перехоплення виключень. 3. Дати короткі рекомендації по створенню власних виключень. 4. Висновки.
8.	<b>Індивідуальне завдання 8:</b> <b>Файли та друк</b> 1. Дати короткий огляд потоків введення-виведення Java 2. Дати опис принципів роботи файлового вводу-виводу. 3. Дати короткі рекомендації по програмуванню друку засобами Java 2D. 4. Висновки.

№ п/п	Теми рефератів
9.	<b>Індивідуальне завдання 9:</b> <b>Багатопотокове програмування.</b> 1. Дати короткий огляд класу Thread і інтерфейсу Runnable. 2. Дати опис принципів реалізації інтерфейсу Runnable 3. Дати короткі рекомендації з використання класу Thread 4. Висновки.

**ТЕСТОВІ ЗАВДАННЯ ДЛЯ САМОКОНТРОЛЮ**

1) Доступ до змінних внутрішнього класу із зовнішнього класу

Варіант відповіді:

- a) Без обмежень
- b) Обмеження з модифікатором *protected*
- c) Обмеження з модифікатором *private*
- d) Заборонений з модифікатором *protected*
- e) Заборонене
- d.

2) Доступ до змінних зовнішнього класу із внутрішнього класу

Варіант відповіді:

- a) Без обмежень
- b) Обмеження з модифікатором *protected*
- c) Обмеження з модифікатором *private*
- d) Заборонений з модифікатором *protected*
- e) Заборонений з модифікатором *private*
- a.

3) Доступ до методів внутрішнього класу із зовнішнього класу

Варіант відповіді:

- a) Без обмежень
- b) Обмеження з модифікатором *protected*
- c) Обмеження з модифікатором *private*
- d) Заборонений з модифікатором *protected*
- e) Заборонений зовсім
- e.

4) Доступ до методів зовнішнього класу із внутрішнього класу

Варіант відповіді:

- a) Без обмежень

b) Обмеження з модифікатором *protected*

c) Обмеження з модифікатором *private*

d) Заборонений з модифікатором *protected*

e) Заборонений з модифікатором *private*

a.

5) Варіант створення класу A

Варіант відповіді:

- a) `class A{}`
- b) `private void class A(int b)`
- c) `public static class A(int b)`
- d) `public static class A(static b)`
- e) `protected void class A(int b)`
- a.

6) Розширення класу A

Варіант відповіді:

- a) `class A extends B`
- b) `private void class A(int b) :: class(B)`
- c) `public static class A(int b) :: public class B`
- d) `public static class A(static b) :: extends class B`
- e) `protected void class A(int b) extends class B`
- a.

7) Правильний модифікатор конструктора класу A

Варіант відповіді:

- a) `void constructor A()`
- b) `public A()`
- c) `public constructor A()`
- d) `protected constructor A`
- e) `extern constructor A`
- b.

8) Синтаксис викликів конструкторів `super` класів у порядку спадкування: A, B, C, D

Варіант відповіді:

- a) `super A, B, C, D`
  - b) `super A; B; C; D;`
  - c) `super A(); super B(); super C(); super D();`
  - d) `super D; C; B; A;`
  - e) `super D; super C; super B; super A;`
- c.

9) Порядок динамічного оголошення об'єктів класу A

Варіант відповіді:

- a) `new class A;`
  - b) `class A = new A();`
  - c) `A = new class A;`
  - d) `A = new class A();`
  - e) `new A = A();`
- b.

10) Правильне використання покажчиків у параметрах методу M

Варіант відповіді:

- a) `void M(int* a, char* b);`
  - b) покажчиків немає
  - c) `void M(*int a, *char b);`
  - d) `void M(int a*, char b*);`
  - e) `void M(int** a, char** b);`
- b.

11) Правильне використання посилань у параметрах методу

Варіант відповіді:

- a) `void M (int a, char b);`
  - b) `void M(& int a, char & b);`
  - c) `void M(& int a, & char b);`
  - d) `void M(int a &, char b &);`
  - e) `void M(int && a, char && b);`
- a.

12) Правильний варіант копіювання посилань a і b класів A і B

Варіант відповіді:

- a) `a -> & b;`
- b) `a = & b;`
- c) `& -> a & b;`
- d) `& a = & b`

e) `a = b;`

e.

13) Правильна форма виведення на консоль повідомлення msg

Варіант відповіді:

- a) `System.out.println (msg);`
  - b) `system.println(msg);`
  - c) `println(msg);`
  - d) `printf(msg);`
  - e) `type(msg);`
- a.

14) Правильне використання ключового слова this

Варіант відповіді:

- a) `this.a = 10;`
  - b) `This.a = 10;`
  - c) `This -> a = 10;`
  - d) `a.this = 10;`
  - e) `a = this 10;`
- a.

15) Призначення методу finalize()

Варіант відповіді:

- a) Для захисту класу від копіювання
  - b) Для захисту класу від спадкування
  - c) Для виконання дій перед видаленням об'єкта
  - d) Для завершення дій об'єкта
  - e) Для виходу із програми
- c.

16) Призначення ключового слова final

Варіант відповіді:

- a) Для захисту змінної від копіювання
  - b) Для захисту змінної від зміни
  - c) Для виконання дій перед видаленням змінної
  - d) Для завершення дій змінної
  - e) Для знищення змінної
- b.

17) Призначення ключового слова `static` для змінної

Варіант відповіді:

- a) Змінна не переміщається в пам'яті
  - b) Змінна може бути збережена на диску
  - c) Змінна стає глобальною
  - d) Змінна стає константою
  - e) Змінна видаляється при виході з методу
- с.

18) Правильне застосування інтерфейсу `ITF`

Варіант відповіді:

- a) `class A extends ITF;`
  - b) `class A extends (ITF);`
  - c) `class A = extends (ITF);`
  - d) `class A = implements (ITF);`
  - e) `class A implements ITF();`
- е.

19) Правильне застосування оператора `#define` у мові Java

Варіант відповіді:

- a) немає
  - b) `#define a = 10`
  - c) `#define a 10`
  - d) `#define a = 10;`
  - e) `#define a 10;`
- а.

20) Правильне застосування функції `printf()` у мові Java для виведення повідомлення `msg`

Варіант відповіді:

- a) `printf(msg);`
  - b) немає
  - c) `system.out.printf(msg);`
  - d) `system.print.printf(msg);`
  - e) `system.console.printf(msg);`
- б.

21) Правильне застосування функції `scanf()` у мові Java для консольного введення

Варіант відповіді:

- a) `int a = system.scnaf();`
  - b) `char a = system.scnaf();`
  - c) немає
  - d) `int a = system.console.scnaf();`
  - e) `int a = scanf();`
- с.

22) Правильне застосування функції `putc()` для виведення символу 'C' на консоль

Варіант відповіді:

- a) `system.out.putc(C);`
  - b) `system.out.putc('C');`
  - c) `system.out.putc("C");`
  - d) немає
  - e) `system.out.putc(\'C');`
- д.

23) Які з оголошень коректні, якщо:

```
class Outer {
    class Inner {
    }
}
```

Варіант відповіді:

- a) `new Outer.Inner();`
  - b) `Outer.new Inner();`
  - c) `new Outer.new Inner();`
  - d) `new Outer().new Inner();`
  - e) `Outer.Inner();`
- д.

24) Які з наступних тверджень дійсні?

Варіант відповіді:

- a) Частні методи не можуть бути перевантажені
  - b) Перевизначений метод не може включати виключення не оброблювані в базовому класі
  - c) Методи, оголошені як `final`, не можуть бути перевизначені
  - d) Статичні методи не можуть бути перевизначені
  - e) Статичні методи оголошені як `final`, можуть бути перевизначені
- с.



## 25) Що відбудеться в результаті компіляції і запуску коду?

```
class Base {}
class A extends Base {}
public class Quest{
    public static void main(String[]
    args){
        Base b = new Base();
        A ob = (A) b;
    }
}
```

Варіант відповіді:

- a) Помилка під час виконання `Base b = new Base();`
- b) Нічого: компіляція і виконання без помилок
- c) Помилка під час компіляції
- d) Системна помилка
- e) Помилка під час виконання `A ob = (A) b;`
- c.

## 26) Який буде результат виконання програми

```
public class Test {
    public static void main(String[]
    args) {
        Test test = new Test();
        int i = 5;
        while(i = 5){
            System.out.println(i++);
        }
    }
}
```

Варіант відповіді:

- a) Компілятор видасть повідомлення про помилку
- b) На консоль будуть послідовно виведені значення 01234
- c) На консоль будуть послідовно виведені значення 43210
- d) Програма відкомпілюється, але на консоль нічого виведено не буде
- e) Об'єкт `test` не буде створений
- a.

## 27) Що буде виведено при спробі компіляції і запуску цієї програми:

```
public class Quest {
    public static void main(String[]
    args){
```

```
int a[] = new int[]{1,2,3,};
System.out.print(a[1]);
    }
}
```

Варіант відповіді:

- a) Помилка компіляції: не визначений розмір масиву
- b) Помилка часу виконання
- c) Виведено 1
- d) Виведено 2
- e) Помилка компіляції: неправильна ініціалізація
- d.

## 28) Що буде виведено при спробі компіляції і запуску програми?

```
public class Quest{
    static int j=2;
    public static void result(int i){
        i *= 10;
        j += 2;
    }
    public static void main(String[]
    args){
        char i = '1';
        result(i);
        System.out.println(i+" "+j);
    }
}
```

Варіант відповіді:

- a) 1 2
- b) 10 2
- c) Помилка: параметр методу `result()` не сполучається з переданою змінною
- d) 10 4
- e) 1 4
- e.

## 29) Що буде виведено при компіляції і запуску коду?

```
public class Quest {
    {System.out.print("1");}
    Static
    {System.out.print("2");}
    Quest()
    {System.out.print("3");}
    public static void main(String[]
    args) {
        System.out.print("4");
    }
}
```

Варіант відповіді:

- a) 34
  - b) 24
  - c) 14
  - d) 4
  - e) 1234
- b.

30) Правильне застосування функції delete() у мові Java для видалення з пам'яті об'єкта A

Варіант відповіді:

- a) delete(A);
  - b) немає
  - c) A -> delete();
  - d) delete(\*A);
  - e) delete(&A);
- b.

31) Бібліотека Swing це:

Варіант відповіді:

- a) Набір класів GUI
  - b) Компоненти GUI
  - c) Методи GUI
  - d) Класи AWT
  - e) Події GUI
- a.

32) Бібліотека Swing має архітектуру

Варіант відповіді:

- a) архітектура JVC
  - b) архітектура DWI
  - c) архітектура WFC
  - d) архітектура MVC
  - e) архітектура Wifi
- d.

33) В основі Swing лежать елементи

Варіант відповіді:

- a) Компоненти
  - b) Контейнери
  - c) Компоненти і контейнери
  - d) Моделі
  - e) Моделі і контейнери
- c.

34) Компонент Swing це:

Варіант відповіді:

- a) Незалежний візуальний елемент керування
  - b) Залежний візуальний елемент керування
  - c) Модель візуального елемента керування
  - d) Програмний елемент керування
  - e) Користувацький елемент керування
- a.

35) До компонентів Swing не належить:

Варіант відповіді:

- a) JColorChooser
  - b) JDialog
  - c) JFrame
  - d) JLayeredPane
  - e) JMenuPane
- e.

36) У Swing не входить компонент:

Варіант відповіді:

- a) JSlider
  - b) JTable
  - c) JToggleButton
  - d) JViewport
  - e) JComboBox
- c.

37) Помилка в написанні компонента Swing:

Варіант відповіді:

- a) JEditorPane
  - b) JInternalFrame
  - c) JList
  - d) JOptionPane
  - e) JProgressBar
- b.

38) Наступний компонент відсутній у бібліотеці Swing:

Варіант відповіді:

- a) JScrollBar
- b) JSpinner
- c) JTextArea

- d) JToolBar
- e) JWindows
- e.

39) Swing має кількість типів контейнерів:

Варіант відповіді:

- a) Один
- b) Два
- c) Три
- d) Чотири
- e) П'ять
- b.

40) Компонент JLabel:

Варіант відповіді:

- a) Пасивний
- b) Активний
- c) Інтерактивний
- d) Інтерпасивний
- e) Користувацький
- a.

41) Компонент JMenu:

Варіант відповіді:

- a) Пасивний
- b) Активний
- c) Інтерактивний
- d) Інтерпасивний
- e) Користувацький
- b.

42) Компонент JRadioButton:

Варіант відповіді:

- a) Пасивний
- b) Активний
- c) Інтерактивний
- d) Інтерпасивний
- e) Користувацький
- b.

43) Компонент JPasswordField:

Варіант відповіді:

- a) Пасивний
- b) Активний
- c) Інтерактивний
- d) Інтерпасивний

- e) Користувацький
- b.

44) Розміри вікна задаються методом:

Варіант відповіді:

- a) `Frame.setSize(int x, int y);`
- b) `setSize(int x, int y);`
- c) `Frame.setDimension(int x, int y);`
- d) `Frame.setSize(int x, int y, int z, int w);`
- e) `Frame.setSize(float x, float y);`
- a.

45) При закритті вікна верхнього рівня потрібно припинити роботу додатка оператором

Варіант відповіді:

- a) `setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);`
- b) `setDefaultCloseOperation(JFrame.CLOSE_ON_EXIT);`
- c) `setDefaultCloseOperation(EXIT_ON_CLOSE);`
- d) `setDefaultCloseOperation(DO_NOTHING_ON_CLOSE);`
- e) `setDefaultCloseOperation(HIDE_ON_CLOSE);`
- a.

46) Компоненти Swing створюються:

Варіант відповіді:

- a) У головному потоці додатка
- b) У спеціальному потоці додатка
- c) У потоці диспетчеризації подій
- d) У потоці диспетчеризації компонентів
- e) У потоці диспетчеризації контейнерів
- c.

47) Для диспетчеризації подій використовується метод:

Варіант відповіді:

- a) `static void invokeLater(Runnable об'єкт)`
- b) `static void invokeAndWait(Event об'єкт)`

- c) `static void invokeLater(Event  
об'єкт)`  
 d) `static void  
invokeAndWait(InterruptedExceptio  
н об'єкт)`  
 e) `static void  
invokeLater(InvocationTargetException  
об'єкт)`

a.

48) Правильна форма визначення слухача події:

Варіант відповіді:

- a) `obj.addActionListener(new  
ActionListener() {...`  
 b) `obj.setActionListener(new  
ActionListener() {...`  
 c) `obj.addActionListener(set  
ActionListener() {...`  
 d) `obj.addActionListener(obj  
ActionListener() {...`

e)

`set.addActionListener(ActionListe  
ner() {...`

a.

49) Правильна форма визначення оброблювача події:

Варіант відповіді:

- a) `public action void  
Performed(ActionEvent ae) {`  
 b) `set actionPerformed(ActionEvent  
ae) {`  
 c) `static void  
actionPerformed(ActionEvent ae) {`  
 d) `public void  
actionPerformed(ActionEvent ae) {`  
 e) `void actionPerformed(ActionEvent  
ae) {`  
 d.

50) Компонент не успадковує клас JComponent:

Варіант відповіді:

- a) JButton  
 b) JTable  
 c) JScrollPane  
 d) JToggleButton  
 e) JTabbedButton  
 e.

51) Неіснуючий клас кнопок Swing:

Варіант відповіді:

- a) JButton  
 b) JToggleButton  
 c) JCheckBox  
 d) JRadioButton  
 e) JCheckButton  
 e.

52) Неправильний конструктор класу JButton:

Варіант відповіді:

- a) JButton(Icon icon)  
 b) JButton(String text)  
 c) JButton(String text, Icon icon)  
 d) JButton(Action a)  
 e) JButton(JIcon icon)  
 e.

53) Неправильний конструктор класу JCheckBox:

Варіант відповіді:

- a) JCheckBox()  
 b) JCheckBox(Action a)  
 c) JCheckBox(Icon icon)  
 d) JCheckBox(String text)  
 e) JCheckBox(String text, Icon icon,  
String selected)  
 e.

54) Неправильний конструктор класу JFrame:

Варіант відповіді:

- a) JFrame()  
 b) JFrame(GraphicsConfiguration gc)  
 c) JFrame(Action a)  
 d) JFrame(String title,  
GraphicsConfiguration gc)  
 e) JFrame(String title)  
 c.

55) Неправильний конструктор класу JMenu:

Варіант відповіді:

- a) JMenu()  
 b) JMenu(String s)  
 c) JMenu(Icon icon)

- d) JMenu(Action a)
- e) JMenu(String s, boolean b)
- c.

56) Неправильний конструктор класу JMenuItem()

Варіант відповіді:

- a) JMenuItem()
- b) JMenuItem(Action a)
- c) JMenuItem(Icon icon)
- d) JMenuItem(Icon icon, String text)
- e) JMenuItem(String text)
- d.

57) Неправильний конструктор класу JPasswordField()

Варіант відповіді:

- a) JPasswordField()
- b) JPasswordField(Document doc, String txt, int columns)
- c) JPasswordField(int columns)
- d) JPasswordField(Action a)
- e) JPasswordField(String text)
- d.

58) Неправильний конструктор класу JSlider()

Варіант відповіді:

- a) JSlider()
- b) JSlider(Action a)
- c) JSlider(int orientation)
- d) JSlider(int min, int max, int value)
- e) JSlider(int orientation, int min, int max, int value)
- b.

59) Неправильний конструктор класу JTable()

Варіант відповіді:

- a) JTable()
- b) JTable(int numRows, int numColumns)
- c) JTable(TableModel dm)
- d) JTable(TableModel dm, TableColumnModel cm)
- e) JTable (Vector rowData, int numColumns)

e.

60) Неправильний конструктор класу JTextArea()

Варіант відповіді:

- a) JTextArea()
- b) JTextArea(Document doc)
- c) JTextArea(int rows, int columns)
- d) TextArea(String text, int rows)
- e) JTextArea(String text)
- d.

61) Неправильний конструктор класу JToggleButton()

Варіант відповіді:

- a) JToggleButton()
- b) JToggleButton(Action a, Icon icon)
- c) JToggleButton(String text)
- d) JToggleButton(Icon icon)
- e) JToggleButton(Action a)
- b.

62) Неправильний конструктор класу JToolBar()

Варіант відповіді:

- a) JToolBar()
- b) JToolBar(int orientation)
- c) JToolBar(String name)
- d) JToolBar(Icon icon, int orientation)
- e) JToolBar(String name, int orientation)
- d.

63) Неправильні константи класу Calendar()

Варіант відповіді:

- a) ERA
- b) YEAR
- c) HOUR\_OF\_DAY
- d) MINUTES
- e) SECOND
- d.

64) Помилка в рядку

Варіант відповіді:

- a) JSlider sl = new JSlider(sl);
- b) sl.setMajorTickSpacing(10);

- c) `sl.setMinorTickSpacing(5);`
- d) `sl.setPaintTicks(true);`
- e) `sl.setPaintLabels(true);`
- a.

65) Виключення. Оператор `throws`

Варіант відповіді:

- a) Збуджує виключення
- b) Перехоплює виключення
- c) Обробляє виключення
- d) Перевіряє виключення
- e) Ігнорує виключення
- a.

66) Виключення. Блок `try {}`

Варіант відповіді:

- a) Збуджує виключення
- b) Інкапсулює виключення
- c) Обробляє виключення
- d) Перевіряє виключення
- e) Ігнорує виключення
- b.

67) Виключення. Оператор `catch()`

Варіант відповіді:

- a) Збуджує виключення
- b) Перехоплює виключення
- c) Обробляє виключення
- d) Перевіряє виключення
- e) Ігнорує виключення
- b.

68) Неправильна назва потоку введення-виведення або методу:

Варіант відповіді:

- a) `stdin`
- b) `stdout`
- c) `stdprint`
- d) `stderr`
- e) `read(in)`
- c.

69) Неправильна назва класу введення-виведення:

Варіант відповіді:

- a) `Reader()`
- b) `Writer()`

- c) `InputStream()`
- d) `OutputStream()`
- e) `WriterStream ()`
- e.

70) Методи, відсутні в класі

`OutputStream()`

Варіант відповіді:

- a) `write(char[] buf)`
- b) `write(char[] buf, int offset, int len)`
- c) `write(int elem)`
- d) `write (int i)`
- e) `write(String s)`
- d.

71) Класи, що не створюють потоки, пов'язані з файлами:

Варіант відповіді:

- a) `FileReader()`
- b) `FileWriter()`
- c) `FileWriterStream ()`
- d) `FileInputStream()`
- e) `FileOutputStream()`
- c.

72) Класи, що не створюють потоки, пов'язані з масивами:

Варіант відповіді:

- a) `CharArrayReader()`
- b) `CharArrayWriter()`
- c) `ByteArrayInputStream()`
- d) `ByteArrayOutputStream()`
- e) `IntArrayWriter ()`
- e.

73) Класи, що не створюють потоки, пов'язані з підпроцесами:

Варіант відповіді:

- a) `PipedReader()`
- b) `PipedReaderStream ()`
- c) `PipedOutputStream()`
- d) `PipedInputStream()`
- e) `PipedWriter()`
- b.

74) Класи, що не створюють буферизовані потоки

Варіант відповіді:

- a) `BufferedReader`
- b) `BufferedReaderStream`
- c) `BufferedWriter`
- d) `BufferedInputStream`
- e) `BufferedOutputStream`
- b.

75) У середовищі потокової багатозадачності найменшим елементом керованого коду є:

Варіант відповіді:

- a) Потік
- b) Модуль
- c) Об'єкт
- d) Файл
- e) Завдання
- a.

76) Більш пріоритетний потік:

Варіант відповіді:

- a) Виконується швидше
- b) Частіше запускається
- c) Займає більше пам'яті
- d) Займає менше пам'яті
- e) Має більше привілеїв
- b.

77) Для створення потоку необхідно:

Варіант відповіді:

- a) Розширити клас `Thread`
- b) Розширити інтерфейс `Runnable`
- c) Розширити клас `Runnable`
- d) Реалізувати інтерфейс `JavaRunnable`
- e) Реалізувати інтерфейс `Thread`
- a.

78) Помилка в імені константи стану потоку:

Варіант відповіді:

- a) `NEW`
- b) `RUNNABLED`
- c) `BLOCKED`
- d) `TERMINATED`
- e) `WAITING`
- b.

79) Неправильний конструктор класу `Thread()`

Варіант відповіді:

- a) `Thread()`
- b) `Thread(Runnable target)`
- c) `Thread(String name)`
- d) `Thread(Runnable target, String name)`
- e) `Thread(Runnable target, ThreadGroup group, String name)`
- e.

80) Правильний варіант оголошення масиву:

Варіант відповіді:

- a) `int ar = new int[];`
- b) `int ar[] = new int;`
- c) `int ar[] = new int[];`
- d) `int ar = new int;`
- e) `int ar[] = new [];`
- c.

81) Застосування покажчиків в Java

Варіант відповіді:

- a) Застосовуються
- b) Застосовуються рідко
- c) Не застосовуються
- d) Застосовуються в певних випадках
- e) Застосовуються завжди
- c.

82) Застосування посилань в Java

Варіант відповіді:

- a) Застосовуються
- b) Застосовуються рідко
- c) Не застосовуються
- d) Застосовуються в певних випадках
- e) Застосовуються завжди
- c.

83) Визначення пакета в Java:

Варіант відповіді:

- a) `package Pack;`
- b) `package = Pack;`

- c) package = new Pack;
- d) new package = Pack;
- e) new package Pack;
- a.

84) Тип даних не є цілочисельним:

Варіант відповіді:

- a) byte
- b) short
- c) int
- d) float
- e) long
- d.

85) Тип даних є цілочисельним:

Варіант відповіді:

- a) float
- b) double
- c) int
- d) unsigned float
- e) unsigned double
- c.

86) Неправильний тип даних:

Варіант відповіді:

- a) float
- b) double
- c) int
- d) char
- e) unsigned int
- e.

87) Розрядність типу byte:

Варіант відповіді:

- a) 1
- b) 4
- c) 8
- d) 12
- e) 16
- c.

88) Розрядність типу int:

Варіант відповіді:

- a) 4
- b) 8
- c) 16
- d) 32

- e) 64
- d.

89) Розрядність типу short:

Варіант відповіді:

- a) 4
- b) 8
- c) 16
- d) 32
- e) 64
- c.

90) Розрядність типу long:

Варіант відповіді:

- a) 4
- b) 8
- c) 16
- d) 32
- e) 64
- e.

91) Розрядність типу float:

Варіант відповіді:

- a) 4
- b) 8
- c) 16
- d) 32
- e) 64
- d.

92) Розрядність типу double:

Варіант відповіді:

- a) 4
- b) 8
- c) 16
- d) 32
- e) 64
- e.

93) Розрядність типу char:

Варіант відповіді:

- a) 4
- b) 8
- c) 16
- d) 32
- e) 64
- c.



94) Правильний варіант приведення типів до byte b:

Варіант відповіді:

- a) `b = (int)i;`
  - b) `b = (byte)i;`
  - c) `b = int i;`
  - d) `b = int(i);`
  - e) `b = byte(i);`
- b.

95) `byte b = 3`. Виникне помилка часу виконання

Варіант відповіді:

- a) `b *= 2;`
  - b) `b *= 4;`
  - c) `b *= 8;`
  - d) `b *= 100;`
  - e) `b += 200;`
- d.

96) Операція інкремента:

Варіант відповіді:

- a) `++`
  - b) `--`
  - c) `+=`
  - d) `*=`
  - e) `/=`
- a.

97) Операція декремента:

Варіант відповіді:

- a) `++`
  - b) `--`
  - c) `+=`
  - d) `*=`
  - e) `/=`
- b.

98) Операція додавання

Варіант відповіді:

- a) `++`
  - b) `--`
  - c) `+=`
  - d) `*=`
  - e) `/=`
- c.

99) Операція ділення

Варіант відповіді:

- a) `++`
  - b) `--`
  - c) `+=`
  - d) `*=`
  - e) `/=`
- e.

100) Операція множення

Варіант відповіді:

- a) `++`
  - b) `--`
  - c) `+=`
  - d) `*=`
  - e) `/=`
- d.

### Шкала оцінювання: національна та ECTS

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою
		для екзамену, курсової роботи
90 – 100	<b>A</b>	відмінно
82-89	<b>B</b>	добре
74-81	<b>C</b>	
64-73	<b>D</b>	задовільно
60-63	<b>E</b>	
35-59	<b>FX</b>	незадовільно з можливістю повторного складання
0-34	<b>F</b>	незадовільно з обов'язковим повторним вивченням дисципліни

### Види самостійної роботи студентів

Репродуктивна самостійна робота	Самостійне прочитання, перегляд, конспектування навчальної літератури, прослуховування лекцій, магнітофонних записів, завчання, переказ, запам'ятовування, Інтернет-ресурси, повторення навчального матеріалу та ін.
Пізнавально-пошукова самостійна робота	Підготовка повідомлень, доповідей, виступів на семінарських і практичних заняттях, добір літератури по дисциплінарних проблемах, написання рефератів, контрольних, курсових робіт і ін.
Творча самостійна робота	Написання рефератів, наукових статей, участь у науково - дослідницькій роботі, підготовка дипломної роботи (проекту). Виконання спеціальних завдань і ін., участь у студентській науковій конференції.

### Організація і контроль самостійної роботи

Для успішного виконання самостійної роботи студентів необхідне планування і контроль з боку викладачів. Аудиторна самостійна робота виконується студентами на лекціях, семінарських заняттях, і, отже, викладач повинен заздалегідь вибудувати систему самостійної роботи, враховуючи всі її форми, мети, відбираючи навчальну і наукову інформацію та засоби (методичних) комунікацій, продумуючи роль студента в цьому процесі та своя участь в ньому.

Питання для самостійної роботи студентів, зазначені в робочій програмі дисципліни, пропонуються викладачами на початку вивчення дисципліни. Студенти мають право вибирати теми, що додатково цікавлять, для самостійної роботи.

Зміст діяльності викладача і студента при виконанні самостійної роботи представлений у таблиці.

## Самостійна робота

Основні характеристики	Діяльність студентів
Ціль виконання СР	<ul style="list-style-type: none"> <li>- Розуміє і приймає мету СР як особистісно значущу</li> <li>- ознайомлюється з вимогами до СР</li> </ul>
Мотивація	<ul style="list-style-type: none"> <li>- Формує власну пізнавальну потребу у виконанні СР;</li> <li>- формує установку і приймає рішення щодо виконання СР</li> </ul>
Керування	На основі володіння узагальненим прийомом сам здійснює управління СР (проектує, планує, раціонально розподіляє час і т.д.)
Контроль і корекція виконання СР	<ul style="list-style-type: none"> <li>- Здійснює поточний операційний самоконтроль за ходом виконання СР;</li> <li>- Виявляє, аналізує і виправляє допущені помилки та вносить корективи в роботу, відстежує хід виконання СР;</li> <li>- Веде пошук оптимальних способів виконання СР;</li> <li>- Здійснює рефлексивне відношення до власної діяльності;</li> <li>- Здійснює підсумковий самоконтроль результату СР</li> </ul>
Оцінка	На основі співвіднесення результату з метою дає самооцінку СР, своїм пізнавальним можливостям, здібностям і якостям

Не применшуючи значення аудиторної самостійної роботи, у даних методичних рекомендаціях акцентується увага на проблемах, пов'язаних з позааудиторною самостійною роботою і її організацією. Позааудиторна самостійна робота студентів (далі самостійна робота) – планована навчальна, учбово-дослідницька, науково-дослідницька діяльність студентів, здійснювана у позааудиторний час за завданням і при методичнім керівництві викладача, але без його особистої участі. Вона містить у собі:

- підготовку до аудиторних занять (лекціям, практичним, семінарським, лабораторним роботам і ін.) і виконання відповідних завдань;
- самостійну роботу над окремими темами навчальних дисциплін відповідно до учбово-тематичних планів;
- написання рефератів, доповідей;
- підготовку до всіх видів практики і виконання передбачених ними завдань;
- виконання письмових контрольних і курсових робіт;
- підготовку до всіх видів контрольних випробувань, у тому числі до комплексних іспитів і залікам;
- підготовку до підсумкової державної атестації, у тому числі виконання випускної кваліфікаційної (дипломної) роботи (проекту);
- роботу в студентських наукових суспільствах, кружках, семінарах та ін.;
- участь у роботі факультативів, семінарів і т.п.;
- участь у науковій і науково-методичній роботі кафедри;
- участь у наукових і науково-практичних конференціях, семінарах, конгресах і т.п.;
- інші види діяльності, організовуваних і здійснюваної вузом, факультетом або кафедрою.

Виконання будь-якого виду самостійної роботи припускає проходження студентами наступних етапів:

- визначення мети самостійної роботи;
- конкретизація пізнавальної (проблемної або практичної) роботи;

- самооцінка готовності до самостійної роботи з розв'язку поставленої або обраного завдання;
- вибір адекватного способу дій, що веде до розв'язку завдання (вибір шляхів і засобів для її розв'язку);
- планування (самостійно або за допомогою викладача) самостійної роботи з розв'язку завдання;
- реалізація програми виконання самостійної роботи.

### **Методичні поради і рекомендації до завдань**

Усі типи завдань, виконуваних студентами, у тому числі в процесі самостійної роботи, так чи інакше містять установку на набуття і закріплення певного обсягу знань, а також на формування в рамках цих знань деяких навичок розумових операцій - вміння оцінювати, аналізувати, порівнювати, коментувати і т.д. Деякі завдання вимагають пояснення:

1. Прокоментувати висловлення - пояснити, яка ідея укладена в уривку, про яку позицію її автора вона свідчить.
2. Зрівняти – виявити подібність і відмінність позицій по певних ознаках.
3. Обґрунтувати один з декількох запропонованих варіантів відповіді – привести аргументи на користь правильності обраного варіанта відповіді й указати, у чому помилковість інших варіантів.
4. Аргументувати (обґрунтувати, довести, пояснити) відповідь – значить:
  - а) виправдати (спростувати) деяку точку зору;
  - б) обґрунтувати свою точку зору, опираючись на теоретичні або практичні узагальнення, дані і т.д.
5. Провести аналіз – розкласти досліджувані явища на складові частини, зіставити їх з метою виявлення в них істотного, необхідного й визначального.

6. Тезисно викласти ідею, концепцію, теорію – використовуючи матеріал навчальних посібників і іншої літератури, коротко, але не на шкоду змісту сформулювати основні положення навчання.
7. Дати характеристику, охарактеризувати явища – значить назвати істотні, необхідні ознаки якого-небудь явища (положення якої-небудь теорії) і виявити особливості.
8. Зобразити схематично – значить розкрити зміст відповіді у вигляді таблиці, малюнка, діаграми й інших графічних форм.

### **Робота з літературою**

Важливою складовою самостійної підготовки є робота з літературою до всіх видів занять: лабораторним, семінарським, практичним, при підготовці до заліків, іспитів, тестування, участі в наукових конференціях.

Уміння працювати з літературою означає навчитися осмислено користуватися джерелами. Перш ніж приступитися до освоєння наукової літератури, рекомендується читання підручників і навчальних посібників.

Існує кілька методів роботи з літературою.

Один з них – найвідоміший – метод повторення: прочитаний текст можна завчити напам'ять. Просте повторення впливає на пам'ять механічно й поверхово. Отримані таким шляхом відомості легко забуваються. Щоб ґрунтовно обробити інформацію й закодувати її для зберігання, важливо зробити цілий ряд розумових операцій:

- прокоментувати нові дані;
- оцінити їхнє значення;
- поставити питання;
- зіставити отримані відомості з раніше відомими.

Для поліпшення обробки інформації дуже важливо встановлювати осмислені зв'язки, структурувати нові відомості.

Вивчення наукової, навчальної та іншої літератури вимагає ведення робочих записів.

Форма записів може бути досить різноманітною: простий або розгорнутий план, тези, цитати, конспект.

План – першооснова, каркас якої-небудь письмової роботи викладу, що визначають послідовність, матеріалу.

План є найбільш короткою і тому самою доступною та розповсюдженою формою записів змісту вихідного джерела інформації. По суті, це перелік основних питань, розглянутих у джерелі. План може бути простим і розгорнутим. Їхня відмінність полягає в ступені деталізації змісту і, відповідно, в обсязі.

Перевага плану полягає в наступному.

- *По-перше*, план дозволяє щонайкраще усвідомити логіку думки автора, спрощує розуміння головних моментів добутку.
- *По-друге*, план дозволяє швидко і глибоко проникнути в сутність побудови добутку і, отже, набагато легше орієнтуватися в його змісті.
- *По-третє*, план дозволяє – при наступному поверненні до нього – швидше звичайного згадати прочитане.
- *По-четверте*, за допомогою плану набагато зручніше відшукувати в джерелі потрібні місця, факти, цитати і т.д.

Виписки – невеликі фрагменти тексту (неповні і повні пропозиції, окремі абзаци, а також дослівні і близькі до дослівних запису про факти, що викладаються в ньому), що містять у собі квінтесенцію змісту прочитаного.

Виписки являють собою більш складну форму записів змісту вихідного джерела інформації. По суті, виписки – не що інше, як цитати, запозичені з тексту. Виписки дозволяють у концентрованій формі і з максимальною точністю відтворити в довільному (частіше послідовному) порядку найбільш важливі думки автора, статистичні і даталогічні відомості. В окремих випадках цілком припустимо замінити цитування викладом, близьким до дослівного.

Тези – стислий виклад змісту вивченого матеріалу в стверджувальній формі.



Відмінність тез від звичайних виписок полягає в наступному.

- *По-перше*, тезам властивий значно більш високий ступінь концентрації матеріалу.
- *По-друге*, у тезах відзначається перевага висновків над загальними міркуваннями.
- *По-третє*, найчастіше тези записуються близько до оригінального тексту, тобто без використання прямого цитування.

Виходячи зі сказаного, неважко виявити основну перевагу тез: вони незамінні для підготовки глибокої та всебічної аргументації письмової роботи будь-якої складності, а також для підготовки виступів на захисті, доповідей та ін.

Анотація – короткий виклад основного змісту вихідного джерела інформації, що дає про нього узагальнене представлення. До написання анотацій прибігають у тих випадках, коли справжня цінність і придатність вихідного джерела інформації виконавцеві письмової роботи остаточно неясна, але в той же час про нього необхідно залишити короткий запис із узагальнюючою характеристикою. Для зазначеної мети й використовується анотація.

Характерною рисою анотації поряд зі стислістю й узагальненістю її змісту є й те, що пишеться анотація завжди після того, як (хоча б попередньо) завершено ознайомлення зі змістом вихідного джерела інформації. Крім того, пишеться анотація майже винятково своїми словами й лише у вкрай рідких випадках містить у собі невеликі витримки оригінального тексту.

Резюме – коротка оцінка вивченого змісту вихідного джерела інформації, отримана, насамперед, на основі висновків, що втримуються в ньому. Резюме досить подібно по своїй суті з анотацією. Однак, на відміну від останньої, текст резюме концентрує в собі дані не з основного змісту вихідного джерела інформації, а з його заключної частини, насамперед висновків. Але, як і у випадку з анотацією, резюме викладається своїми словами – витримки з оригінального тексту в ньому практично не зустрічаються.

Конспект – складний запис змісту вихідного тексту, що включає в себе запозичення (цитати) найбільш примітних місць у комбінації із планом джерела, а також стислий аналіз записаного матеріалу і висновки по ньому.

*Для роботи над конспектом впливає:*

- визначити структуру конспектуємого матеріалу, чому значною мірою сприяє письмове ведення плану по ходу вивчення оригінального тексту;
- у відповідності зі структурою конспекту зробити відбір і наступний запис найбільш істотного змісту оригінального тексту - у формі цитат або у викладі, близькому до оригіналу;
- виконати аналіз записів і на його основі – доповнення записів власними зауваженнями, міркуваннями, "фактурою", запозиченої з інших джерел і т.п. (розташовувати все це прямує на полях зошита для записів або на окремих аркушах-вкладках);
- завершити формулювання і запис висновків по кожній із частин оригінального тексту, а також загальних висновків.

Систематизація вивчених джерел дозволяє підвищити ефективність їх аналізу та узагальнення. Підсумком цієї роботи повинна стати логічно вибудована система відомостей по суті досліджуваного питання.

Необхідно із усього матеріалу виділити існуючі точки зору на проблему, проаналізувати їх, зрівняти, дати їм оцінку.

До речі, цій процедурі повинні зазнати й матеріали з Інтернету щоб уникнути механічного скачування готових текстів. У записах і конспектах студентові дуже важливо вказувати назви джерел, авторів, рік видання. Це організує його, а головне, знадобиться в наступному навчанні. Безумовно, студент повинен брати за правило активно працювати з літературою в інших, бібліотеках, використовуючи, у тому числі, їх комп'ютерні можливості (електронна бібліотека в мережі Інтернет).

## Реферат

Реферат – короткий виклад змісту документа або його частини, наукової роботи, що включає основні фактичні відомості і висновки, необхідні для первісного ознайомлення із джерелами і визначення доцільності звертання до них.

Сучасні вимоги до реферату – точність і об'єктивність у передачі відомостей, повнота відображення основних елементів як по змісту, так і за формою.

Ціль реферату - не тільки повідомити про зміст роботи, але і дати представлення про знову виниклі проблеми відповідної галузі науки.

У навчальному процесі реферат являє собою короткий виклад у письмовій формі або у формі публічної доповіді змісту книги, навчання, наукового дослідження й т.п.

Інакше кажучи, це доповідь на певну тему, що висвітлює її питання на основі огляду літератури та інших джерел.

Реферати в рамках навчального процесу у вузі оцінюються по наступним основним критеріями:

- актуальність змісту, високий теоретичний рівень, глибина і повнота аналізу фактів, явищ, проблем, що ставляться до теми;
- інформаційна насиченість, новизна, оригінальність викладу питань;
- простота і дохідливість викладу;
- структурна організованість, логічність, граматична правильність і стилістична виразність;
- переконливість, аргументованість, практична значимість і теоретична обґрунтованість пропозицій і висновків.

Складання списку використаної літератури. Відповідно до вимог, пропонованих до реферату, доповіді, необхідно скласти список літератури, використаної в роботі над ним.

## **Основні етапи роботи над рефератом**

В організаційному плані написання реферату - процес, розподілений у часі по етапах. Усі етапи роботи можуть бути згруповані в три основні: підготовчий, виконавський і заключний.

Підготовчий етап містить у собі пошуки літератури по певній темі з використанням різних бібліографічних джерел; вибір літератури в конкретній бібліотеці; визначення кола довідкових посібників для наступної роботи з теми.

Виконавський етап містить у собі читання книг (інших джерел), ведення записів прочитаного.

Заключний етап містить у собі обробку наявних матеріалів і написання реферату, складання списку використаної літератури.

Написання реферату. Визначений список літератури по темі реферату. Вивчена історія питання по різних джерелах, складені виписки, довідки, плани, тези, конспекти. Первісне завдання даного етапу - систематизація і переробка знань. Систематизувати отриманий матеріал - значить привести його в певний порядок, який відповідав би наміченому плану роботи.

## **Структура реферату**

### Вступ

Вступ - це вступна частина реферату, що випереджає текст.

Він повинне містити наступні елементи:

- а) дуже короткий аналіз наукових, експериментальних або практичних досягнень у тій області, якої присвячений реферат;
- б) загальний огляд опублікованих робіт, розглянутих у рефераті;
- в) мета даної роботи;
- г) завдання, що вимагають розв'язки.

Обсяг вступу при обсязі реферату 10-15 може становити одну сторінку.

### Основна частина .

В основній частині реферату студент дає письмовий виклад матеріалу за запропонованим планом, використовуючи матеріал із джерел. У цьому розділі роботи формулюються основні поняття, їх зміст, підходи до аналізу, що існують у літературі, точки зору на суть проблеми, її характеристики.

Відповідно до поставленого завдання робляться висновки і узагальнення. Дуже важливо не повторювати, не копіювати стиль джерел, а виробити свій власний, який відповідає характеру матеріалу.

### Висновок

Висновок підбиває підсумок роботи. Він може включати повтор основних тез роботи, щоб акцентувати на них увага читачів (слухачів), містити загальний висновок, до якого прийшов автор реферату, пропозиції по подальшій науковій розробці питання й т.п. Тут уже ніякі конкретні випадки, факти, цифри не аналізуються. Висновок за обсягом, як правило, повинен бути менше вступу.

### **Список використаних джерел**

У строго алфавітному порядку розміщаються всі джерела, незалежно від форми і змісту: офіційні матеріали, монографії та енциклопедії, книги і документи, журнали, брошури та газетні статті.

Список використаних джерел оформляється в тій же послідовності, яка зазначена у вимогах до оформлення рефератів, курсових, дипломних робіт.

### **Порядок здачі і захисту рефератів.**

1. Реферат здається на перевірку викладачеві
2. При захисті реферату викладач враховує:
  - якість;
  - ступінь самостійності студента і виявлену ініціативу;
  - зв'язність, логічність і грамотність складання;
  - оформлення відповідно до вимог.
3. Захист тематичного реферату може проводитися в рамках годин навчальної дисципліни або конференції або по одному реферату при вивченні відповідної теми, або за домовленістю з викладачем.
4. Захист реферату студентом передбачає
  - доповідь по реферату не більш 5-7 хвилин
  - відповіді на запитання опонента.

На захисті заборонене читання тексту реферату.

### Рекомендована література

1. Кроссплатформені мови програмування: Методичні вказівки до виконання лабораторних робіт для студентів денної форми навчання за спеціальністю 123 «Комп'ютерна інженерія» / В.В. Смірнов, Н.В. Смірнова. - Кропивницький : ЦНТУ.
2. Конспект лекцій з дисципліни «Кроссплатформені мови програмування» / В.В. Смірнов, Н.В. Смірнова. - Кропивницький : ЦНТУ.
3. Кроссплатформені мови програмування: Методичні вказівки до виконання самостійних робіт для студентів денної форми навчання за спеціальністю 123 «Комп'ютерна інженерія» / В.В. Смірнов, Н.В. Смірнова. - Кропивницький : ЦНТУ.
4. Шілдіт Герберт Java. Повне керівництво, 8-е вид. / Шілдіт Герберт - К.: 2012. - 1104 с.
5. Шілдіт Герберт Swing: посібник для початківців / Шілдіт Герберт - К.: 2007. – 704 с.
6. Еккель Брюс Філософія Java. Бібліотека програміста, 4-те вид / Еккель Брюс - К.: 2009. - 640 с.
7. Хорстманн Кей С., Java. Бібліотека професіонала/Том 1. Основи. 9-е вид./Хорстманн Кей С., Корнелл Гарі - К.: 2014. - 864 с.
8. Хорстманн Кей С., Java. Бібліотека фахівця / Том 2. Розширені кошти. 9-те вид. / Хорстманн Кей С., Корнелл Гарі - К.: 2014. - 1008 с.
9. [www.sgmu.ru/info/str/metod/files/cab/instr/mr\\_selfws.rtf](http://www.sgmu.ru/info/str/metod/files/cab/instr/mr_selfws.rtf)
10. [guap.ru/guap/kaf84old63/meth/sam\\_rab\\_ilinskaya.pdf](http://guap.ru/guap/kaf84old63/meth/sam_rab_ilinskaya.pdf) Методичні рекомендації щодо виконання самостійної роботи студентів / професор, д.е.н. Є.М. Іллінська / 2011
11. <http://www.oracle.com/technetwork/java/javase/downloads/jdk-7-netbeans-download-432126.html>