

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

Кафедра систем штучного інтелекту



**Лабораторна робота №5
з дисципліни
“Об’єктно-орієнтоване програмування”**

**Виконав:
студент групи КН-107
Антоник Анастасія-Марія
Старший викладач:
Гасько Р.Т.**

Львів 2018

Test

(6/6 балів)

1. Чи буде зкомпільовано наступний клас?

```
class Test
{
    int test(int i, int d)
    {
        return 0;
    }

    static int test(int i, double d)
    {
        return 0;
    }

    static double test(double i, double d)
    {
        return 0;
    }
}
```

☒ Так ✓

☐ Ні

2. Скільки конструкторів матиме наступний клас? Виберіть найбільш точну відповідь

```
public class A
{
}
```

☒ Один ✓

☐ Жодного

☐ Такий клас викличе помилку компіляції

3. Чи буде зкомпільовано наступний інтерфейс?

```
interface A
{
    private int i;
}
```

☐ Так

☒ Ні ✓

4. Вкажіть причину помилки компіляції для наступного коду. Виберіть найбільш точну відповідь

```
interface A
{
    void test();
}

class B implements A
{
    void test()
    {
        System.out.println("Ok");
    }
}
```

☐ метод test() в класі B повинен бути static

☐ метод test() в класі B повинен бути final

☒ метод test() в класі B повинен бути public ✓

☐ тут не буде помилки компіляції

5. Клас було оголошено як абстрактний без жодного абстрактного методу. Чи буде він зкомпільований? Виберіть найбільш точну відповідь

```
abstract class Demo
{
}
```

☒ так ✓

☐ так, за умови що буде додано абстрактний конструктор

☐ ні

6. Наступний код викликає помилку при компіляції. Вкажіть причину.

```
abstract class A
{
    abstract int demo();
}

class B extends A
{
}
```

☐ Клас B не містить конструктора

☐ Клас A не містить абстрактного конструктора

☒ Клас B не реалізував всі абстрактні методи батьківського класу ✓

7. Чи буде зкомпільовано наступний код? Виберіть найбільш точну відповідь

```
interface A
{
    int variable = 1;
}

class B implements A
{
    void methodB()
    {
        variable = 2;
    }
}
```

☐ так

☐ ні, оскільки variable вже оголошено

☒ ні, оскільки variable вже оголошено і це константа ✓

8. Чи буде зкомпільовано наступний код?

```
class A
{
}

interface Demo extends A
{
}
```

☐ так

☐ ні, оскільки A не реалізує інший інтерфейс

☒ ні, оскільки інтерфейс не може розширювати клас ✓

9. Що буде виведено на екран в результаті виконання наступного коду?

```
abstract class Calculator
{
    abstract int sum(int a, int b);
}

public class Demo
{
    public static void main(String[] args)
    {
        int result = new Calculator()
        {
            int sum(int a, int b)
            {
                return a+b;
            }
        }.sum(2, 3);
        System.out.println(result);
    }
}
```

☐ Нічого

☐ Помилка часу виконання

☒ 5 ✓

☐ 6

10. Що буде виведено на екран в результаті виконання наступного коду?

```
interface Iface1
{
    String a = "A";
    String methodA();
}

interface Iface2 extends Iface1
{
    String b = "B";
    String methodB();
}

class Demo implements Iface1, Iface2
{
    public String methodA()
    {
        return a+b;
    }

    public String methodB()
    {
        return b+a;
    }
}

public class Main
{
    public static void main(String[] args)
    {
        Demo demo = new Demo();
        System.out.println(demo.methodA());
        System.out.println(demo.methodB());
    }
}
```



AB

BA



BA

AB



Помилка часу виконання



Помилка компіляції

11. Що буде виведено на екран в результаті виконання наступного коду?

```
class Demo
{
    void test(String s){
        System.out.println("String");
    }

    void test(Integer i){
        System.out.println("Integer");
    }
}

public class Main
{
    public static void main(String[] args)
    {
        Demo demo = new Demo();
        demo.test(null);
    }
}
```

☐ Integer

☐ String

☐ null

☒ нічого, помилка компіляції ✓

12. Чи дозволено викликати this() та super() в методах?

☐ так

☒ ні ✓

Application.java Command.java datenow.java echo.java exit.java

```
1 package com.tasks5.command;
2
3
4 public class Application {
5     public static void main(String[] args){
6         if (args==null || args.length>2 || args.length==0)
7         {
8             System.out.println("Error");
9             return;}
10        if (args[0].equals("help")) {
11            new help().execute();
12        }
13        else if (args[0].equals("help")&& args.length == 1) {
14            new exit().execute();
15        }
16        else if(args[0].equals("date") && args[1].equals("now") ) {
17            new datenow().execute();
18        }
19        else if(args[0].equals("echo") && args.length == 2) {
20            new echo(args[1]).execute();
21        }
22        else {
23            System.out.println("Error");
24        }
25    }
26 }
```

Application.java Command.java datenow.java echo.java

```
1 package com.tasks5.command;
2
3 public class datenow implements Command{
4     public void execute(){
5         System.out.println(System.currentTimeMillis());
6     }
7 }
```



```

Application.java  echo.java  Command.java
1 package com.tasks5.command;
2
3 public class echo implements Command {
4     private String p;
5     public echo() {
6
7     }
8     public echo(String p){
9         this.p = p;
10    }
11
12    public void execute(){
13        System.out.println(p);
14    }
15 }

```

```

Application.java  exit.java  help.java  ec
1 package com.tasks5.command;
2
3 public class exit implements Command {
4     public void execute(){
5         System.out.println("Goodbye!");
6     }
7 }

```

```

Application.java  exit.java  help.java  echo.java
1 package com.tasks5.command;
2
3 public class help implements Command {
4     public void execute(){
5         System.out.println("Help executed");
6     }
7 }

```

Висновок: на цій лабораторній роботі я дізналась про інтерфейси, зрозуміла для чого вони, та як їх застосовувати. Написала програму, у якій є класи, що імплементуються від інтерфейсу.