

Lab 01 Assignment

班级: 202111

学号: 19241027

姓名: 胡峻诚

Question01

(a) 上述源文件的名字是什么?

- Person.java

(b) 编译上述源文件将生成几个字节码文件? 这些字节码文件的名字分别是什么?

- 2个字节码文件, 名称分别为 Person.class Xiti.class

(c) 在命令行执行 java Person 得到怎样的错误提示? 执行 java xiti 得到怎样的错误提示? 执行 java Xiti.class 得到怎样的错误提示? 执行 java Xiti 得到怎样的输出结果?

- 执行 java Person

```
[base] huakaiyiji@huakaiyiji-yeluoyide-MacBook-Pro src % java Person
错误: 在类 Person 中找不到 main 方法, 请将 main 方法定义为:
    public static void main(String[] args)
否则 JavaFX 应用程序类必须扩展 javafx.application.Application
```

- 执行 java xiti

```
[base] huakaiyiji@huakaiyiji-yeluoyide-MacBook-Pro src % java xiti
错误: 找不到或无法加载主类 xiti
原因: java.lang.NoClassDefFoundError: Xiti (wrong name: xiti)
```

- 执行 java Xiti.class

```
[base] huakaiyiji@huakaiyiji-yeluoyide-MacBook-Pro src % java Xiti.clas
s
错误: 找不到或无法加载主类 Xiti.class
原因: java.lang.ClassNotFoundException: Xiti.class
```

- 执行 java Xiti

```
[base] huakaiyiji@huakaiyiji-yeluoyide-MacBook-Pro src % java Xiti
您好, 很高兴认识您 nice to meet you
```

Question02

```

public class Name {
    public static void main(String[] args){
        System.out.println("19241027 胡峻诚");
    }
}

```

Question03

```

public class Unicode {
    public static void main(String args[]) {
        char c = '大';
        System.out.println("[" + c + "] 的Unicode编码: " + (int) c);
        int num = 23398;
        System.out.println("Unicode编码为" + num + "的字符是: " + (char) num);
    }
}

```

Question04

The screenshot shows the IntelliJ IDEA interface with the project 'Lab01' open. The 'src' directory contains several Java files: Person.java, Name.java, Unicode.java, UseIntArray.java, and TwoDimensionArray.java. The 'UseIntArray.java' file is currently selected and displayed in the editor. The code defines a main method that initializes an integer array with values 8, 9, 12, 13, and 14, calculates its sum, and prints the result. The 'Run' tool window shows the output of the program, which is:

```

intArray[0]=8
intArray[1]=9
intArray[2]=12
intArray[3]=13
intArray[4]=14

SUM=56

```

The status bar at the bottom indicates the build was successful in 695 milliseconds.

Question05

The screenshot shows the IntelliJ IDEA interface with the project 'Lab01' open. The 'src' directory contains several Java files: Person.java, Name.java, Unicode.java, UseIntArray.java, and TwoDimensionArray.java. The 'TwoDimensionArray.java' file is currently selected and displayed in the editor. The code defines a main method that creates a 3x4 integer array 'b', initializes it with specific values, and then creates three 3x3 arrays 'arr1', 'arr2', and 'arr3' using different assignment methods. The 'Run' tool window shows the output of the program, which is:

```

b[0][0]=1000
sum=1139
b.length=3
arr1:
0 1 2 3 4 5 6 7 8 9 10 11
arr2:
12
13
14
15
16
17
18
19
20
21
22
23
arr3:
0 1 2 3 4 5 6 7 8

```

The status bar at the bottom indicates the build was successful in 695 milliseconds.

Question06

The screenshot shows the IntelliJ IDEA interface with the project 'Lab01' open. The current file is 'E.java'. The code prints the English alphabet in a specific pattern:

```

public class E {
    public static void main (String args[ ]) {
        char c = '\0';
        for(int i=1;i<=4;i++) {
            switch(i) {
                case 1: c = 'J';
                System.out.print(c);
                case 2: c = 'e';
                System.out.print(c);
                break;
                case 3: c = 'p';
                System.out.print(c);
                default: System.out.print("好");
            }
        }
    }
}

```

The output window shows the printed characters: J, e, p, 好.

Question07

The screenshot shows the IntelliJ IDEA interface with the project 'Lab01' open. The current file is 'Square.java'. The code prints a square pattern of numbers:

```

public class Square {
    public static void main(String[] args) {
        int n = Integer.valueOf(args[0]);
        int cnt = 1;
        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
                System.out.print(cnt++ + " ");
            }
            System.out.println();
        }
    }
}

```

The terminal window shows the output for n=5 and n=7:

```

Last login: Sun Mar 13 22:04:08 on ttys001
(base) huakaijijielyuoyide@huakaijiji-yeluooyide-MacBook-Pro src % javac Square.java
(base) huakaijijielyuoyide@huakaijiji-yeluooyide-MacBook-Pro src % java Square 5
1 2 3 4 5
6 7 8 9 10
11 12 13 14 15
16 17 18 19 20
21 22 23 24 25
(base) huakaijijielyuoyide@huakaijiji-yeluooyide-MacBook-Pro src % java Square 7
1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31 32 33 34 35
36 37 38 39 40 41 42
43 44 45 46 47 48 49
(base) huakaijijielyuoyide@huakaijiji-yeluooyide-MacBook-Pro src %

```

Question08

The screenshot shows the IntelliJ IDEA interface with the project 'Lab01' open. The 'Star.java' file is the active editor. The code prints a star pattern to the console. A terminal window on the right shows the execution of the Java compiler and the resulting star pattern.

```

public class Star {
    public static void main(String[] args) {
        int n = Integer.valueOf(args[0]);
        int line = 0;
        for (; line <= n / 2; line++) {
            for (int i = 0; i < (n - 2 * line) / 2; i++)
                System.out.print(" ");
            for (int j = 0; j < 2 * line + 1; j++)
                System.out.print("*");
            System.out.println();
        }
        for (; line < n; line++) {
            for (int i = 0; i < (2 * line - n) / 2 + 1; i++)
                System.out.print(" ");
            for (int j = 0; j < 2 * (n - line) - 1; j++)
                System.out.print("*");
            System.out.println();
        }
    }
}

```

```

src -- zsh - 80x24
Last login: Sun Mar 13 22:09:12 on ttys001
(base) huakaiyijielyuoyide@huakaiyiji-yeluoyide-MacBook-Pro src % javac Star.java
(base) huakaiyijielyuoyide@huakaiyiji-yeluoyide-MacBook-Pro src % java Star 5
*****
*****
*****
*****
*****
*****
*****
*****
(base) huakaiyijielyuoyide@huakaiyiji-yeluoyide-MacBook-Pro src % java Star 7
*****
*****
*****
*****
*****
*****
*****
*****
(base) huakaiyijielyuoyide@huakaiyiji-yeluoyide-MacBook-Pro src %

```

Question09

The screenshot shows the IntelliJ IDEA interface with the project 'Lab01' open. The 'GetPi.java' file is the active editor. The code calculates pi using a series. A terminal window on the right shows the execution of the Java compiler and the resulting value of pi.

```

import java.util.Scanner;

public class GetPi {
    public static double getPi(int n) {
        if (n <= 0) n = 0;
        double sum = 0;
        for (int i = 0; i <= n; i++) {
            sum = ((i & 1) == 1 ? sum - 1.0 / (2 * i + 1) : sum + 1.0 / (2 * i + 1));
        }
        return 4 * sum;
    }

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        int n = input.nextInt();
        System.out.println(getPi(n));
    }
}

```

```

/Library/Java/JavaVirtualMachines/jdk-17.0.2.jdk
100/100
3.1415936535887745
进程已结束,退出代码0

```

Question10

The screenshot shows the IntelliJ IDEA interface with the project 'Lab01' open. The 'src' directory contains several Java files. The current file is 'Qsort.java', which contains the implementation of a quick sort algorithm. The code includes three static methods: `main`, `qsort`, and `qsort1`. The `qsort` method is annotated with a yellow warning icon. The output window shows the sorted array [1, 2, 3, 4, 5, 6, 7, 8].

```

public class Qsort {
    public static void main(String[] args) {
        int arr[] = {3, 5, 4, 6, 1, 2, 8, 7};
        System.out.println(arr.length);
        qsort(arr);
        for (int i = 0; i < arr.length; i++) {
            System.out.print(arr[i] + " ");
        }
    }

    public static int[] qsort(int arr[]) {
        if (arr == null)
            return null;
        return qsort(arr, start: 0, end: arr.length - 1);
    }

    public static int[] qsort(int arr[], int start, int end) {
        int x = arr[start];
        int i = start;
        int j = end;
        while (i < j) {
            while (i < j && arr[j] > x)
                j--;
            while (i < j && arr[i] < x)
                i++;
            if (arr[i] == arr[j] && i < j)
                i++;
            else {
                int temp = arr[i];
                arr[i] = arr[j];
                arr[j] = temp;
            }
        }
        if (i - 1 > start) arr = qsort(arr, start, end: i - 1);
        return arr;
    }
}

```

Question13

The screenshot shows the IntelliJ IDEA interface with the project 'Lab01' open. The 'src' directory contains several Java files. The current file is 'Sort.java', which implements a search algorithm. The code imports `java.util.Arrays` and `java.util.Scanner`. It sorts an array and then searches for a value `x` using a for loop. The output window shows the sorted array [-45, 12, 45, 67, 67, 89, 123] and the message '12 is in the array.'

```

import java.util.Arrays;
import java.util.Scanner;

public class Sort {
    public static void main(String[] args) {
        int arr[] = {12, 45, 67, 89, 123, -45, 67};
        Arrays.sort(arr);
        for (int i = 0; i < arr.length; i++) {
            System.out.print(arr[i] + " ");
        }
        System.out.println("");
        Scanner input = new Scanner(System.in);
        int x = input.nextInt();
        boolean have = false;
        for (int i = 0; i < arr.length; i++) {
            if (arr[i] == x) {
                have = true;
                break;
            }
        }
        if (have) {
            System.out.println(x + " is in the array.");
        } else {
            System.out.println(x + " is not in the array.");
        }
    }
}

```

Question 11

```

public class ForInString {
    public static void main(String[] args) {
        for (char c : "An African Swallow".toCharArray())
            System.out.print(c + " ");
    }
}

```

The output window shows the printed string: /Library/Java/JavaVirtualMachines/jdk-17.0.2.jdk/Cont
An African Swallow

Question12

如图所示：

```

class Point {
    int x,y;
    void setXY(int m,int n){
        x = m;
        y = n;
    }
}

public class Example {
    public static void main(String args[]) {
        Point p1,p2;
        p1=new Point();
        p2=new Point();
        System.out.println("p1的引用:"+p1);
        System.out.println("p2的引用:"+p2);
        p1.setXY( m: 1111, n: 2222 );
        p2.setXY( m: -100, n: -200 );
        System.out.println("p1的x,y坐标:"+p1.x+","+p1.y); // 1
        System.out.println("p2的x,y坐标:"+p2.x+","+p2.y); // 2
        p1 = p2;
        p1.setXY( m: 0, n: 0 );
        System.out.println("p1的引用:"+p1);
        System.out.println("p2的引用:"+p2);
        System.out.println("p1的x,y坐标:"+p1.x+","+p1.y); // 3
        System.out.println("p2的x,y坐标:"+p2.x+","+p2.y); // 4
    }
}

```

The output window shows the printed values:

```

p1的引用:Point@65b3120a
p2的引用:Point@6f539caf
p1的x,y坐标:1111,2222
p2的x,y坐标:-100,-200
p1的引用:Point@6f539caf
p2的引用:Point@6f539caf
p1的x,y坐标:0,0
p2的x,y坐标:0,0

```

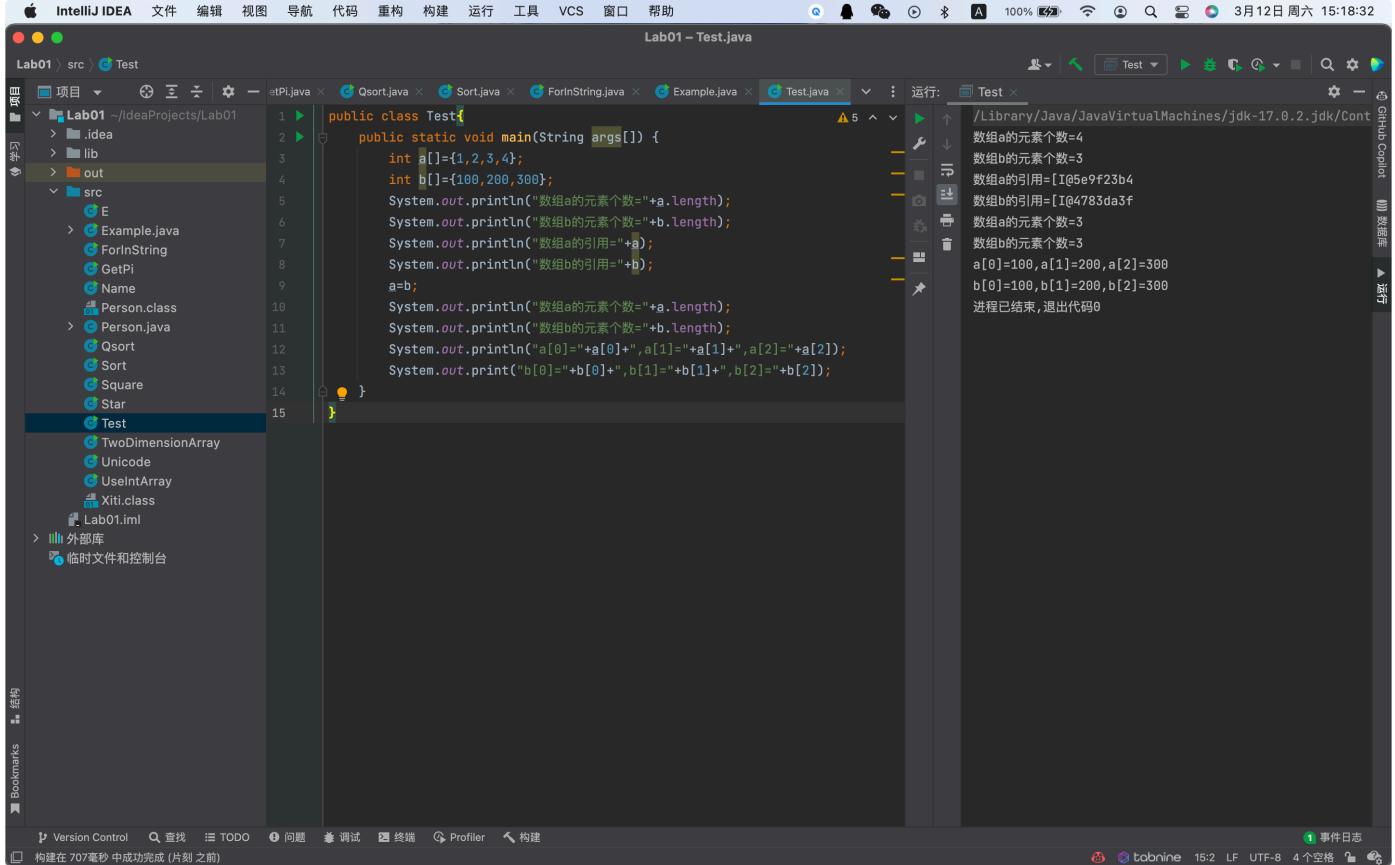
进程已结束,退出代码0

- 浅拷贝（同一个引用）：被复制对象的所有变量都含有与原来的对象相同的值，而所有的对其他对象的引用仍然指向原来

的对象。换言之，浅复制仅仅复制所考虑的对象，而不复制它所引用的对象。

- 深拷贝(新引用):被复制对象的所有变量都含有与原来的对象相同的值，除去那些引用其他对象的变量。那些引用其他对象的变量将指向被复制过的新对象，而不再是原有的那些被引用的对象。换言之，深复制把要复制的对象所引用的对象都复制了一遍。

Question13



The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Structure:** The project is named "Lab01". The "src" directory contains several Java files: etPi.java, Qsort.java, Sort.java, ForInString.java, Example.java, and Test.java. Test.java is the active file.
- Code Editor:** The code in Test.java demonstrates array copying. It creates two arrays, a and b, with elements {1, 2, 3, 4} and {100, 200, 300} respectively. It then prints the lengths of both arrays and their respective references. Finally, it prints the elements of both arrays.
- Run Output:** The output window shows the results of the program execution. It prints:
 - 数组a的元素个数=4
 - 数组b的元素个数=3
 - 数组a的引用=[I@6e9f23b4
 - 数组b的引用=[I@4783da3f
 - 数组a的元素个数=3
 - 数组b的元素个数=3
 - a[0]=100,a[1]=200,a[2]=300
 - b[0]=100,b[1]=200,b[2]=300
 A message at the end says "进程已结束,退出代码0".
- Bottom Bar:** Shows tabs for Version Control, Search, TODO, Problems, Profiler, and Build. It also indicates a successful build ("构建在 707毫秒 中成功完成 (片刻之前)").

Question14

```

public class E2 {
    public static void main(String args[]) {
        f(...x: 1,2);
        f(...x: -1,-2,-3,-4); //给参数传值时, 实参的个数很灵活
        f(...x: 9,7,6) ;
    }
    public static void f(int ... x){ //x是可变参数的代表, 代表若干个int型参数
        for(int i=0;i<x.length;i++) { //x.length是x代表的参数的个数
            System.out.println(x[i]); //x[i]是x代表的第一个参数(类似数组)
        }
    }
}

```

进程已结束,退出代码0

Question15

```

public class Strscat {
    public static void main(String[] args){
        System.out.println(strscat(...args: "a", "b", "c", "", "e"));
    }
    public static String strscat(String... args){
        String newString = "";
        for(String str : args){
            newString += str;
        }
        return newString;
    }
}

```

进程已结束,退出代码0

调用 `strscat(new String[]{"a", "b"})` 能通过编译吗?

- 可以

如果还有静态方法 `String strscat(String[] args)` 同时存在，代码能编译通过吗？这时传入的参数是什么？又会返回什么？

- 不能

如果我们声明的是 `String strscat(String[] args)`，`strscat()` 这样的调用还能通过编译吗？如果能，这时传入参数和返回值的情况和 `String strscat(String... args)` 通过相同吗？

- 可以。返回值和之前相同。

Question16

```

public class LoseWeight {
    String name;
    double weight;
    public void Person(String name, double weight) {
        this.name = name;
        this.weight = weight;
    }
    public void loseWeight(double changeWeight) { this.weight += changeWeight; }
    public void printWeight(){
        System.out.println(this.name + "'s weight is " + this.weight);
    }
}
class Lose {
    public static void main(String[] args){
        LoseWeight person = new LoseWeight();
        person.Person(name: "xiaowang", weight: 70);
        person.loseWeight( changeWeight: 45-70);
        person.printWeight();
    }
}

```

Question17

The screenshot shows the IntelliJ IDEA interface with the following details:

- File Menu:** 文件 (File), 编辑 (Edit), 视图 (View), 导航 (Navigate), 重构 (Refactor), 运行 (Run), 工具 (Tools), VCS (Version Control), 窗口 (Windows), 帮助 (Help).
- Toolbar:** Standard toolbar items.
- Status Bar:** 100% battery, 3月12日 周六 16:13:44 (March 12, Saturday 16:13:44).
- Project Structure:** 显示了Lab01项目下的src目录，包含Dinner.java, Test.java, LoseWeight.java等文件。
- Code Editor:** 正在编辑Dinner.java文件，内容如下：

```
public class Dinner {
    public static void main(String[] args) {
        Buyer a = new Buyer();
        Buyer b = new Buyer();
        Buyer c = new Buyer();
        a.person( name: "姑姑", price: 0 );
        b.person( name: "妈妈", price: 0 );
        c.person( name: "你", price: 0 );

        Cooker d = new Cooker();
        Cooker e = new Cooker();
        d.person( name: "爸爸" );
        e.person( name: "婶婶" );

        a.buy( what: "萝卜", cost: 10 );
        a.total();
        d.cook( what: "萝卜" );
    }

    class Buyer {
        String name;
        double price;

        public void person(String name, double price) {
            this.name = name;
            this.price = price;
        }

        public void buy(String what, double cost) {
            System.out.println(name + " bought " + what + ", which costs " + cost + ".");
            price += cost;
        }
    }
}
```

- Run Tab:** 显示了运行结果：

- /Library/Java/JavaVirtualMachines/jdk-17
- 姑姑 bought 萝卜, which costs 10.0.
- 姑姑 totally used 10.0.
- 爸爸 cooked 萝卜.

- Bottom Status:** 构建在 609毫秒 中成功完成 (片刻之前)