

比特就业课105期Java方向笔试强训48天day09_10月19日-王世国-测评结果

考生信息 存在作弊行为

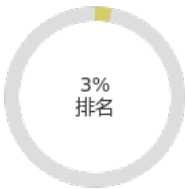


王世国
投递编号：91 | 学校：武汉轻工大学 | 邮箱：1477649017@qq.com | 职位：比特就业课105期Java2班 |
参考区域: 湖北省武汉市 (183.94.121.184) |
做题用时：03:22:19(2022-10-18 20:14:36开始答题，2022-10-19 11:02:55交卷) | 作答设备：PC |
已同意诚信声明和隐私协议

95.0 分 / 100分

在本次考试中，考生总成绩为95.0分/100分，评级为A（排名前3%），编程能力优秀（2题完全通过，分数排名前1%），编程思路完全正确，编程规范性高。该考生在本次考试中存在作弊行为，无视频监控截图，未开启摄像头。

考生成绩



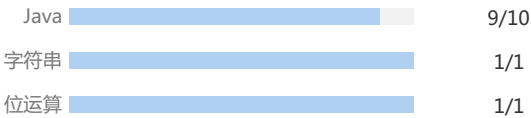
题型	得分	正确题数	排名	用时	是否阅卷
单选	45.0	9	4	00:08:21	已阅
编程	50.0	2	1	01:29:47	已阅

作弊风险

高风险 未开启摄像头

考生考试过程中未开启摄像头

知识点技能图谱



知识点	得分	正确题数
Java	45.0	9
字符串	25.0	1
位运算	25.0	1

历史笔试记录

序号	试卷名称	排名	总得分	得分详情	作弊嫌疑	安排笔试时间	交卷时间
----	------	----	-----	------	------	--------	------

序号	试卷名称	排名	总得分	得分详情	作弊嫌疑	安排笔试时间	交卷时间
1	比特就业课105期+2022寒假班C1考试	20.0%	48.8/60	单选:30.0分 编程:18.75分	否	2022-03-29 11:16:18	2022-03-31 18:51:27
2	比特就业课105期+2022寒假班C2考试	66.0%	24.0/60	单选:24.0分 编程:0.0分	否	2022-04-11 14:12:23	2022-04-11 20:12:16
3	比特就业课 105期JavaSE考试	11.0%	56.0/60	单选:26.0分 编程:30.0分	否	2022-07-12 16:00:16	2022-07-13 15:48:42
4	比特就业课 105期java方向 数据结构考试	47.0%	50.0/60	单选:20.0分 编程:30.0分	否	2022-07-23 12:49:22	2022-07-25 09:56:28
5	比特就业课105期Java方向笔试强训48天 day01_10月10日	36.0%	80.0/100	单选:40.0分 编程:40.0分	是，摄像头监控异常	2022-10-09 17:29:16	2022-10-09 21:57:25
6	比特就业课105期Java方向笔试强训48天 day02_10月11日	13.0%	90.0/100	单选:40.0分 编程:50.0分	是，代码抄袭	2022-10-10 10:43:48	2022-10-10 21:13:15
7	比特就业课105期Java方向笔试强训48天 day03_10月12日	1.0%	95.0/100	单选:45.0分 编程:50.0分	是，摄像头监控异常	2022-10-11 10:40:53	2022-10-12 10:03:09
8	比特就业课105期Java方向笔试强训48天 day04_10月13日	2.0%	95.0/100	单选:30.0分 不定项选择:15.0分 编程:50.0分	是，摄像头监控异常	2022-10-12 10:31:10	2022-10-12 21:14:05
9	比特就业课105期Java方向笔试强训48天 day05_10月14日	12.0%	85.0/100	单选:35.0分 不定项选择:0.0分 编程:50.0分	是，摄像头监控异常 代码抄袭	2022-10-13 11:41:43	2022-10-14 11:33:24
10	比特就业课105期Java方向笔试强训48天 day06_10月15日	13.0%	86.7/100	单选:25.0分 不定项选择:11.67分 编程:50.0分	是，摄像头监控异常	2022-10-14 10:59:38	2022-10-14 21:18:11
11	比特就业课105期Java方向笔试强训48天 day07_10月17日	1.0%	100.0/100	单选:50.0分 编程:50.0分	是，摄像头监控异常	2022-10-16 16:46:53	2022-10-16 20:38:39
12	比特就业课105期Java方向笔试强训48天 day08_10月18日	33.0%	87.5/100	单选:40.0分 编程:47.5分	是，摄像头监控异常	2022-10-17 16:18:42	2022-10-17 20:16:45

编码能力

题号	正确性	提交次数	做题用时	使用语言	运行时间	占用内存	编程思路	代码规范	成绩排名
编程题1	100%	4	00:38:08	Java	54ms	11032K	良	良	1%
编程题2	100%	4	00:51:39	Java	50ms	10932K	良	良	1%

1 [单选题 | 平均分4.36分 | 96人正确/110人做题 | 用时：<1分] 得分：5.0 / 5.0
下面程序的输出是:()

```
String x="fmn";
x.toUpperCase();
String y=x.replace('f','F');
y=y+"wxy";
System.out.println(y);
```

A FmNwxy

- B fmnwxy
- C wxyfmn
- D Fmnwxy

他的回答： D (正确)

正确答案： D

2 [单选题 | 平均分3.29分 | 73人正确/111人做题 | 用时：<1分 | 得分： 5.0 / 5.0

以下多线程对int型变量x的操作,哪个不需要进行同步()

- A ++x
- B x=y
- C x++
- D x=1

他的回答： D (正确)

正确答案： D

3 [单选题 | 平均分3.02分 | 67人正确/111人做题 | 用时：<1分 | 得分： 5.0 / 5.0

关于 JAVA 堆,下面说法错误的是()

- A 所有类的实例和数组都是在堆上分配内存的
- B 对象所占的堆内存是由自动内存管理系统回收
- C 堆内存由存活和死亡的对象,空闲碎片区组成
- D 数组是分配在栈中的

他的回答： D (正确)

正确答案： D

4 [单选题 | 平均分3.41分 | 75人正确/110人做题 | 用时：<1分 | 得分： 5.0 / 5.0

一个以“.java”为后缀的源文件

- A 只能包含一个public类,类名必须与文件名相同
- B 只能包含与文件名相同的public类以及其中的内部类
- C 只能有一个与文件名相同的public类,可以包含其他非public类(不考虑内部类)
- D 可以包含任意public类

他的回答： C (正确)

正确答案： C

5 [单选题 | 平均分2.86分 | 63人正确/110人做题 | 用时：<1分 | 得分： 5.0 / 5.0

JAVA语言的下面几种数组复制方法中,哪个效率最高?

- A for循环逐一复制
- B System.arraycopy
- C Arrays.copyOf
- D 使用clone方法

他的回答： B (正确)

正确答案： B

6 [单选题 | 平均分3.24分 | 72人正确/111人做题 | 用时：<1分 | 得分： 0.0 / 5.0

下面那些情况可以终止当前线程的运行?

- A 当一个优先级高的线程进入就绪状态时

抛出异常如果不捕获处理的话，那就程序直接终止了

B 抛出一个异常时

C 当该线程调用sleep()方法时 sleep方法只是让线程休眠一会儿

D 当创建一个新线程时

他的回答：C (错误)

正确答案：B

7 [单选题 | 平均分2.66分 | 59人正确/111人做题 | 用时：<1分 | 得分：5.0 / 5.0

对文件名为Test.java的java代码描述正确的是()

```
class Person {
    String name = "No name";
    public Person(String nm) {
        name = nm;
    }
}
class Employee extends Person {
    String empID = "0000";
    public Employee(String id) {
        empID = id;
    }
}
public class Test {
    public static void main(String args[]) {
        Employee e = new Employee("123");
        System.out.println(e.empID);
    }
}
```

A 输出：0000

B 输出：123

C 编译报错

D 输出：No name

他的回答：C (正确)

正确答案：C

8 [单选题 | 平均分2.97分 | 66人正确/111人做题 | 用时：<1分 | 得分：5.0 / 5.0

有关下述Java代码描述正确的选项是_____。

```
public class TestClass {
    private static void testMethod(){
        System.out.println("testMethod");
    }
    public static void main(String[] args) {
        ((TestClass)null).testMethod();
    }
}
```

A 编译不通过

B 编译通过，运行异常，报NullPointerException

C 编译通过，运行异常，报IllegalArgumentException

D 编译通过，运行异常，报NoSuchMethodException

E 编译通过，运行异常，报Exception

F 运行正常，输出testMethod

他的回答：F (正确)

正确答案：F

9 [单选题 | 平均分3.95分 | 87人正确/110人做题 | 用时 : <1分] 得分 : 5.0 / 5.0

下列java程序的输出结果为_____。

```
public class Example{
    String str=new String("hello");
    char[]ch={'a','b'};
    public static void main(String args[]){
        Example ex=new Example();
        ex.change(ex.str,ex.ch);
        System.out.print(ex.str+" and ");
        System.out.print(ex.ch);
    }
    public void change(String str,char ch[]){
        str="test ok";
        ch[0]='c';
    }
}
```

- A hello and ab
- B hello and cb
- C hello and a
- D test ok and ab
- E test ok and cb
- F test ok and c

他的回答 : B (正确)

正确答案 : B

10 [单选题 | 平均分2.23分 | 49人正确/110人做题 | 用时 : <1分] 得分 : 5.0 / 5.0

在jdk1.5之后，下列 java 程序输出结果为_____。

```
int i=0;
Integer j = new Integer(0);
System.out.println(i==j);
System.out.println(j.equals(i));
```

Integer类的equals方法其实就是比较的值是否是一样的

- A true,false
- B true,true
- C false,true
- D false,false
- E 对于不同的环境结果不同
- F 程序无法执行

他的回答 : B (正确)

正确答案 : B

11 完善核心代码 [语言限制] [编程题 | 平均分17.57分 | 78人正确/111人做题 | 提交: 3 次] 得分 : 25.0 / 25.0

标题 : 另类加法 | 时间限制 : 3秒 | 内存限制 : 32768K | 语言限制 : [Python, C++, C#, Java]

【另类加法】

给定两个int A和B。编写一个函数返回A+B的值，但不得使用+或其他算数运算符。

测试样例：

1,2

返回：3

代码片段									
功能实现			代码提交统计				代码执行统计		
总通过率	TA的	平均	使用语言	TA的	平均	编译错误	：3		
	100%	70%		Java			答案正确	：1	
	1/1 (100%)	70%		做题用时	00:38:08		00:19:53		
基本测试用例通过率			提交次数	4	2				
代码效率						代码规范及可读性			
	TA的	参考							
运行时间	54ms	3s							
占用内存	11032K	32768K							
						代码规范得分4.4			
						Line 4: Abbreviation in name 'addAB' must contain no more than '1' capital letters. [AbbreviationAsWordInName]			
						Line 4:26: Parameter name 'A' must match pattern '^[a-z][a-z0-9][a-zA-Z0-9]*\$'. [ParameterName]			
						Line 4:33: Parameter name 'B' must match pattern '^[a-z][a-z0-9][a-zA-Z0-9]*\$'. [ParameterName]			

他的代码：

做题用时: 38 分钟 语言: Java 运行时间: 54ms 占用内存: 11032K 程序状态: 答案正确

```
import java.util.*;

public class UnusualAdd {
    public int addAB(int A, int B) {
        // write code here
        while(B != 0){
            int tmp1 = A^B;
            int tmp2 = (A&B)<<1;
            A = tmp1;
            B = tmp2;
        }
        return A;
    }
}
```



[点此](#)或手机扫描二维码查看代码编写过程

12 ACM编程题 语言限制 [编程题 | 平均分15.47分 | 62人正确/108人做题 | 提交: 4 次] 得分: 25.0 / 25.0

标题：走方格的方案数 | 时间限制：1秒 | 内存限制：32768K | 语言限制：不限

【走方格的方案数】

请计算 $n \times m$ 的棋盘格子（ n 为横向的格子数， m 为竖向的格子数）从棋盘左上角出发沿着边缘线从左上角走到右下角，总共多少种走法，要求不能走回头路，即：只能往右和往下走，不能往左和往上走。

注：沿棋盘格之间的边缘线行走

$1 \leq n, m \leq 8$

数据范围：

输入描述：

输入两个正整数n和m，用空格隔开。(1≤n,m≤8)

输出描述：

输出一行结果

示例1：

输入
2
2
输出
6

代码片段

功能实现			代码提交统计			代码执行统计	
总通过率	TA的 100%	平均 61%	使用语言	TA的 Java	平均	答案错误	: 2
基本测试用例通过率	12/12 (100%)	63%	做题用时	00:51:39	00:24:15	答案正确	: 2
边缘测试用例通过率	8/8 (100%)	59%	提交次数	4	2		
代码效率			代码规范及可读性				
运行时间	TA的 50ms	参考 1s	代码规范得分				
占用内存	10932K	32768K	Line 2: 'CLASS_DEF' should be separated from previous statement. [EmptyLineSeparator]				
			Line 6:13: Local variable name 'n' must match pattern '^[a-z][a-z0-9][a-zA-Z0-9]*\$'. [LocalVariableName]				
			Line 7:13: Local variable name 'm' must match pattern '^[a-z][a-z0-9][a-zA-Z0-9]*\$'. [LocalVariableName]				
			4.4				

他的代码：

做题用时: 51 分钟 语言: Java 运行时间: 50ms 占用内存: 10932K 程序状态: 答案正确

```
import java.util.*;
public class Main {
    public static void main(String[] args){
        Scanner scan = new Scanner(System.in);
        //注意这个是棋盘，按点位走的
        int n = scan.nextInt() + 1;//n行就有n+1个点位
        int m = scan.nextInt() + 1;//m行就有m+1个点位
        int[][] arr = new int[n][m];
        for(int i = 0;i < n;i++){
            arr[i][0] = 1;
        }
        for(int i = 1;i < m;i++){
            arr[0][i] = 1;
```

```
}  
for(int i = 1;i < n;i++){  
    for(int j = 1;j < m;j++){  
        arr[i][j] = arr[i][j-1] + arr[i-1][j];  
    }  
}  
System.out.println(arr[n-1][m-1]);  
}  
}
```



[点此](#)或手机扫描二维码查看代码编写过程

监控截图
