

Java方向每日一题day08_12月1日-晁曷恺-测评结果

考生信息



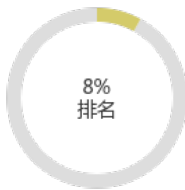
晁曷恺

投递编号：24 | 学校：西北民族大学 | 邮箱：1004945427@qq.com | 职位：2020大四春招冲刺班 |

参考区域：甘肃省兰州市（42.91.68.0） | 做题用时：00:21:55(2020-11-30 22:33:25开始答题，22:55:25交卷) |

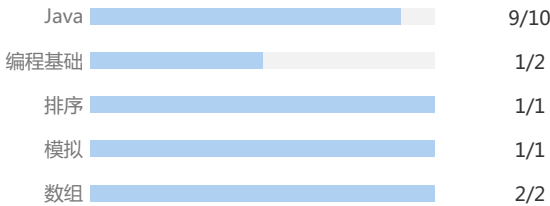
作答设备：PC | 已同意诚信声明和隐私协议

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	14	00:05:49	--
编程	50.0	2	1	00:15:33	--

知识点技能图谱



知识点	得分	正确题数
Java	45.0	9
编程基础	5.0	1
排序	25.0	1
模拟	25.0	1
数组	30.0	2

历史笔记录

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
1	2021届Java方向秋季模拟笔试题-比特职读会	33.0%	57.0/100	单选:42.0分 编程:15.0分	否	2020-08-10 16:24:19	2020-08-10 21:09:17
2	大四春招冲刺班JavaSE考试	2.0%	60.0/60	单选:30.0分 编程:30.0分	否	2020-10-28 17:33:17	2020-10-29 09:24:40
3	Java方向每日一题day01_11月23日	1.0%	100.0/100	单选:50.0分 编程:50.0分	否	2020-11-22 11:26:51	2020-11-23 15:54:01
4	大四春招冲刺班数据结构考试	2.0%	58.0/60	单选:28.0分 编程:30.0分	否	2020-11-23 11:55:15	2020-11-24 09:34:50
5	Java方向每日一题day02_11月24日	1.0%	100.0/100	单选:50.0分 编程:50.0分	否	2020-11-23 12:10:19	2020-11-23 22:55:44

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
6	Java方向每日一题day03_11月25日	25.0%	86.67/100	单选:45.0分 编程:41.67分	否	2020-11-24 15:19:25	2020-11-25 19:17:37
7	Java方向每日一题day04_11月26日	1.0%	100.0/100	单选:50.0分 编程:50.0分	否	2020-11-25 14:58:20	2020-11-25 19:40:27
8	Java方向每日一题day05_11月27日	26.0%	80.0/100	单选:30.0分 编程:50.0分	否	2020-11-25 15:54:38	2020-11-26 23:00:58
9	Java方向每日一题day06_11月28日	19.0%	85.0/100	单选:35.0分 编程:50.0分	否	2020-11-27 14:19:26	2020-11-29 13:39:52
10	Java方向每日一题day07_11月30日	1.0%	100.0/100	单选:50.0分 编程:50.0分	否	2020-11-29 13:58:19	2020-11-30 22:11:44

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%	1	00:07:16	Java	35ms	10760K			1%
编程题2	100%	1	00:08:17	Java	111ms	16044K			1%

1 [平均分3.4分 | 80人正确/119人做题 | 用时：<1分] 得分：5.0 / 5.0

java中关于内存回收的正确说法是

- A 程序员必须创建一个线程来释放内存
- B 内存回收程序负责释放无用内存
- C 内存回收程序允许程序员直接释放内存
- D 内存回收程序可以在指定的时间释放内存对象

他的回答： B (正确)

正确答案： B

2 [平均分4.5分 | 109人正确/120人做题 | 用时：<1分] 得分：5.0 / 5.0

A派生出子类B，B派生出子类C，并且在java源代码中有如下声明：

```
A a0=new A();
A a1=new B();
A a2=new C();
```

以下哪个说法是正确的？

- A 第1行，第2行和第3行的声明都是正确的
- B 第1,2,3行都能通过编译，但第2,3行运行时出错
- C 第1,2行能通过编译，但第3行编译出错
- D 只有第1行能通过编译

他的回答： A (正确)

正确答案： A

3 [平均分4.0分 | 96人正确/119人做题 | 用时 : <1分] 得分 : 5.0 / 5.0

以下说法错误的是 ()

- A 数组是一个对象
- B 数组不是一种原生类
- C 数组的大小可以任意改变
- D 在Java中，数组存储在堆中连续内存空间里

他的回答 : C (正确)

正确答案 : C

4 [平均分1.9分 | 46人正确/120人做题 | 用时 : 3分] 得分 : 0.0 / 5.0

Test.main() 函数执行后的输出是 ()

```
public class Test {
    public static void main(String [] args){
        System.out.println(new B().getValue());
    }
    static class A{
        protected int value;
        public A(int v) {
            setValue(v);
        }
        public void setValue(int value){
            this.value = value;
        }
        public int getValue(){
            try{
                value++;
                return value;
            } catch (Exception e){
                System.out.println(e.toString());
            } finally {
                this.setValue(value);
                System.out.println(value);
            }
            return value;
        }
    }
    static class B extends A{
        public B() {
            super(5);
            setValue(getValue() - 3);
        }
        public void setValue(int value){
            super.setValue(2 * value);
        }
    }
}
```

- A 11 17 34
- B 22 74 74
- C 6 7 7
- D 22 34 17

他的回答 : C (错误)

正确答案：D

5 [平均分4.8分 | 115人正确/120人做题 | 用时：<1分  得分：5.0 / 5.0

关于Java的异常处理机制的叙述哪些正确？

- A 如果程序发生错误及捕捉到异常情况了，才会执行finally部分
- B 其他选项都不正确
- C 当try区段的程序发生异常且被catch捕捉到时，才会执行catch区段的程序
- D catch部分捕捉到异常情况时，才会执行finally部分

他的回答：C (正确)

正确答案：C

6 [平均分4.0分 | 95人正确/120人做题 | 用时：<1分  得分：5.0 / 5.0

如何跳出Array的forEach循环？（ ）

- A break
- B return true
- C return false
- D 以上都不是

他的回答：A (正确)

正确答案：A

7 [平均分4.5分 | 108人正确/119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 只能包含一个类，类名必须与文件名相同
- B 只能包含与文件名相同的类以及其中的内部类
- C 只能有一个与文件名相同的类，可以包含其他类
- D 可以包含任意类

他的回答：C (正确)

正确答案：C

8 [平均分4.5分 | 107人正确/120人做题 | 用时：<1分  得分：5.0 / 5.0

如下Java语句

```
double x= 3.0;
int y=5;
x/=-y;
```

执行后，x的值是（ ）

- A 3
- B 0.6
- C 0.4
- D 0.75

他的回答：D (正确)

正确答案：D

9 [平均分4.7分 | 112人正确/120人做题 | 用时：<1分] 得分：5.0 / 5.0

```
public class Test{
    static int cnt = 6;
    static{
        cnt += 9;
    }
    public static void main(String[] args ) {
        System.out.println ( "cnt =" + cnt);
    }
    static{
        cnt /=3;
    };
}
```

cnt的值是

- A cnt=5
- B cnt=2
- C cnt=3
- D cnt=6

他的回答：A (正确)

正确答案：A

10 [平均分4.3分 | 103人正确/120人做题 | 用时：<1分] 得分：5.0 / 5.0

JUnit主要用来完成什么

- A 发送HTTP请求
- B 建立TCP连接
- C 集成测试
- D 单元测试

他的回答：D (正确)

正确答案：D

11 ACM编程题 语言限制 [平均分19.9分 | 82人正确/103人做题 | 提交: 1次] 得分：25.0 / 25.0

标题：删数 | 时间限制：1秒 | 内存限制：32768K | 语言限制：不限

【删数】有一个数组a[N]顺序存放0~N-1，要求每隔两个数删掉一个数，到末尾时循环至开头继续进行，求最后一个被删掉的数的原始下标位置。以8个数(N=7)为例: { 0, 1, 2, 3, 4, 5, 6, 7 }，0->1->2(删除)->3->4->5(删除)->6->7->0(删除),如此循环直到最后一个数被删除。

输入描述：

每组数据为一行一个整数n(小于等于1000)，为数组成员数,如果大于1000，则对a[999]进行计算。

输出描述：

一行输出最后一个被删掉的数的原始下标位置。

示例1：

输入

8

输出

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 79%	使用语言		TA的 Java	平均		答案正确 : 1
基本测试用例通过率		6/6 (100%)	79%	做题用时		00:07:16	00:48:34		
边缘测试用例通过率		4/4 (100%)	79%	提交次数		1	5		
代码效率						代码规范及可读性			
运行时间		TA的 35ms	参考 1s	代码规范得分 4.30233					
占用内存		10760K	32768K	Line 8:17: Local variable name 'n' must match pattern '^[a-z][a-z0-9][a-zA-Z0-9]*\$'. [LocalVariableName]					
						Line 17:30: '(' is preceded with whitespace. [MethodParamPad]			
						Line 17:35: Parameter name 'n' must match pattern '^[a-z][a-z0-9][a-zA-Z0-9]*\$'. [ParameterName]			

他的代码：

做题用时: 7 分钟 语言：Java 运行时间：35ms 占用内存：10760K 程序状态：答案正确

```
import java.util.*;

public class Main{
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        while (input.hasNextInt()) {
            int n = input.nextInt();

            int res = count(n);
            System.out.println(res);
        }

        input.close();
    }

    private static int count (int n) {
        Queue<Integer> queue = new LinkedList<>();

        if (n >= 1000) {
            n = 1000;
        }

        int num = 0;
        for (int i = 0; i < n; i++) {
            if (num != 2) {
                queue.offer(i);
            }
            num = (num + 1) % 3;
        }

        while (queue.size() > 1) {
            for (int size = queue.size(); size > 0; size--) {
                if (num != 2) {
```

```
        queue.offer(queue.poll());
    } else {
        queue.poll();
    }
    num = (num + 1) % 3;
}
}
return queue.peek();
}
}
```



[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 [平均分20.4分 | 91人正确/112人做题 | 提交: 1 次] 得分: 25.0 / 25.0

标题: n个数里最小的k个 | 时间限制: 1秒 | 内存限制: 32768K | 语言限制: 不限

【n个数里最小的k个】找出n个数里最小的k个

输入描述:

每个测试输入包含空格分割的n+1个整数，最后一个整数为k值,n
不超过100。

输出描述:

输出n个整数里最小的k个数。升序输出

示例1:

输入

3 9 6 8 -10 7 -11 19 30 12 23 5

输出

-11 -10 3 6 7

代码片段					
功能实现			代码提交统计		代码执行统计
	TA的	平均		TA的	答案正确 : 1
总通过率	100%	81%	使用语言	Java	
基本测试用例通过率	2/2 (100%)	81%	做题用时	00:08:17 00:31:44	
			提交次数	1 4	
边缘测试用例通过率	1/1 (100%)	82%			
代码效率				代码规范及可读性	

TA的	参考	代码规范得分	4.26829
运行时间	111ms	1s	Line 10:17: Local variable name 'k' must match pattern '^[a-z][a-z0-9][a-zA-Z0-9]*\$'. [LocalVariableName]
占用内存	16044K	32768K	Line 25:26: Method name 'kMin' must match pattern '^[a-z][a-z0-9][a-zA-Z0-9_]*\$'. [MethodName]
			Line 25:47: Parameter name 'k' must match pattern '^[a-z][a-z0-9][a-zA-Z0-9]*\$'. [ParameterName]

他的代码：

做题用时: 8 分钟 语言: Java 运行时间: 111ms 占用内存: 16044K 程序状态: 答案正确

```
import java.util.*;

public class Main{
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        while (input.hasNextLine()) {
            String[] strs = input.nextLine().split(" ");
            int[] nums = new int[strs.length - 1];
            int k = Integer.parseInt(strs[strs.length - 1]);

            for (int i = 0; i < nums.length; i++) {
                nums[i] = Integer.parseInt(strs[i]);
            }

            int[] res = kMin(nums, k);
            for (int i = 0; i < k; i++) {
                System.out.print(res[i] + " ");
            }
        }

        input.close();
    }

    private static int[] kMin(int[] nums, int k) {
        int[] res = new int[k];
        PriorityQueue<Integer> queue = new PriorityQueue<>((o1, o2) -> o2 - o1);

        for (int i = 0; i < nums.length; i++) {
            queue.offer(nums[i]);
            if (queue.size() > k) {
                queue.poll();
            }
        }

        for (int i = k - 1; i >= 0; i--) {
            res[i] = queue.poll();
        }

        return res;
    }
}
```




[点此](#)或手机扫描二维码查看代码编写过程