



2  
得分



1 / 32  
答对题目数



0 分钟  
用时



前 88 %  
排名

单选题

1. 对一个文件的访问，常由（ ）共同限制

☒ A 用户访问权限和文件属性

☐ B 用户访问权限和文件优先级

☐ C 优先级和文件属性

☐ D 文件属性和口令

正确答案：A 你的答案：C

官方解析：暂无官方题目解析，去讨论区看看吧！

题友讨论(3) ▼

单选题

2. 下列关于TCP和UDP的描述正确的是( )。

☐ A TCP和UDP都是无连接的

☒ B TCP是面向连接的，UDP是面向无连接的

☐ C TCP适用于可靠性较差的广域网，UDP适用于可靠性较高的局域网

☐ D TCP适用于可靠性较高的局域网，UDP适用于可靠性较差的广域网

正确答案：B

官方解析：暂无官方题目解析，去讨论区看看吧！

题友讨论(9) ▼

单选题

3. 存在若干个字符串，若要查找具有相同前缀的字符串，以下哪种数据结构比较适合

☐ A 红黑树

B 哈希表

C Trie树

D 栈

正确答案: C 你的答案: 未作答

官方解析: 暂无官方题目解析, 去讨论区看看吧!

题友讨论(1) ▼

单选题

4. 以下哪个算法是用于求解两个正整数的最大公约数的算法?

A Dijkstra算法

B 辗转相除法

C Floyd算法

D 其他

正确答案: B 你的答案: 未作答

官方解析: 暂无官方题目解析, 去讨论区看看吧!

题友讨论(5) ▲

推荐 最新 楼层

在这里输入你的观点吧



帅固 青出于蓝

C++

这题放在五年级你肯定会

11 回复 发布于 2022-05-27 23:08



不会还有人没offer吧 飞黄腾达 回复

啊哈哈哈哈哈哈

点赞 回复 发布于 2022-06-07 17:31



牛客808469503号 出师牛 回复

狠狠地赞了

点赞 回复 发布于 2022-10-14 09:30 云南



TPH-BETTER. 出师牛

Java

又名欧几里德算法 (Euclidean algorithm), 是求最大公约数的一种方法。它的具体做法是: 用较大数除以较小数, 再用出现的余数 (第一余数) 去除除数, 再用出现的余数 (第二余数) 去除第一余数, 如此反复, 直到最后余数是0为止。如果是求两个数的最大公约数, 那么最后的除数就是这两个数的最大公约数。



讲文明的少年想当offer收割机, 飞黄腾达

门头沟学院 计算机类

哈哈哈, 大学生不会了。

👍 2    💬 回复    发布于 02-21 11:14    四川    来自Android客户端

收起 ▲

单选题

5. 以下哪个数据结构可用来抽象在电影院售票厅排队买票的场景?

☐ A 栈

☐ B 堆

☒ C 队列

☐ D 二叉树

正确答案: C    你的答案: 未作答

官方解析: 暂无官方题目解析, 去讨论区看看吧!

题友讨论(1) ▼

单选题

6. skiplist的查询时间复杂度和以下哪种数据结构不相同?

☐ A 红黑树

☐ B AVL树

☒ C 有序的单链表

☐ D 有序数组

正确答案: C    你的答案: 未作答

官方解析: 暂无官方题目解析, 去讨论区看看吧!

题友讨论(2) ▼

单选题

7. 一个台阶总共有10级, 一次可以向上走1级, 也可以向上走2级, 请问一共有多少种走法?

☐ A 88

☒ B 89

C 90

D 91

正确答案: B 你的答案: 未作答

官方解析: 求斐波那契数列的第n项,  $n=10$

#### 题友讨论(2) ▲

推荐 最新 楼层

在这里输入你的观点吧



弥钵

C++

基础动态规划题, 其实就是斐波那契数列

👍 2 🗨 回复 发布于 2022-02-23 00:19



牛客173915447号 回复

1,2,3,5,8,13,21,34,55,89

👍 点赞 🗨 回复 发布于 2022-04-10 10:58 来自iOS客户端

收起 ▲

[单选题](#)

#### 8. mysql的数据库索引使用的是下面那种数据结构

A skiplist

B 红黑树

C AVL树

D B+树

正确答案: D 你的答案: 未作答

官方解析: 暂无官方题目解析, 去讨论区看看吧!

#### 题友讨论(7) ▲

推荐 最新 楼层

在这里输入你的观点吧



牛客811582976号

数据运营

b

👍 1 🗨 回复 发布于 2022-02-14 15:31 来自Android客户端



牛客378117799号

门头沟学院

**AVL树**: 是一颗二叉平衡树, 左边节点比根节点小, 右边节点比根节点大。左右子树高度差不超过1。

**红黑树**: 是在AVL上提出来的, 是一棵二叉搜索树, 增加了节点颜色, 最长路径不超过最短路径的两倍。对于要求严格的AVL树来说, 它的旋转次数较少, 插入最多两次, 删除最多三次。所以插入删除情况较多时, 用红黑树。

... [展开](#) ✓



zxcv0112358 大橘已定

网易有道 QA

直接从《高性能mysql》的5.1.1索引的类型 搬运了

... 展开

3 回复 发布于 2022-02-20 11:53



牛客184739658号 飞黄腾达

数据分析师

D

我们平常所说的索引，如果没有特别指明，都是指B树(多路搜索树，并不一定是二叉的)结构组织的索引。其中聚集索引，次要索引，覆盖索引，

... 展开

点赞 回复 发布于 2022-02-22 21:38



牛客972826616号 小白牛

重庆移通学院 计算机类

d

点赞 回复 发布于 2022-03-29 21:30 来自Android客户端



牛客95164745号 萌萌牛

测试开发

d

点赞 回复 发布于 2022-04-10 19:11 来自iOS客户端



牛客147582009号 进阶牛

首都师范大学 测绘类

什么是红黑树：

[https://mp.weixin.qq.com/s/X3zYwQXxq93P\\_XUzFmKluQ](https://mp.weixin.qq.com/s/X3zYwQXxq93P_XUzFmKluQ)

点赞 回复 发布于 2022-08-25 13:59 北京

收起

单选题

9. 下面说法正确的是？

- ☒ A epoll ET模式必须配合non-blocking IO使用
- ☐ B epoll LT模式必须配合non-blocking IO使用
- ☐ C epoll ET可以配合blocking IO使用

正确答案：A 你的答案：未作答

官方解析：暂无官方题目解析，去讨论区看看吧！

题友讨论(5)

单选题

10. 在一个空目录下执行umask 333; touch hello;命令后，hello文件的权限为？

☒ A r--r--r--

☐ B rw-rw-rw

☐ C -wx-wx-wx

☐ D rwxrwxrw-

正确答案：A 你的答案：未作答

官方解析：暂无官方题目解析，去讨论区看看吧！

题友讨论(7) ▼

单选题

11. 在DNS系统测试时，假设named进程号是53，如何通知进程重读配置文件

☐ A kill -USR2 53

☐ B kill -USR1 53

☐ C kill -INT 53

☒ D kill -HUP 53

正确答案：D 你的答案：未作答

官方解析：暂无官方题目解析，去讨论区看看吧！

题友讨论(3) ▼

单选题

12. 视图可用于

☐ A 保存存储在一个单独表中的数据的额外副本

☒ B 限制对表中特定行或列中的数据的访问

☐ C 在从基础表中删除历史数据前将其保存

正确答案：B 你的答案：未作答

官方解析：暂无官方题目解析，去讨论区看看吧！

题友讨论(2) ▼

单选题

### 13. 要添加索引的一个理由是

- ☐ A 减少存储空间
- ☐ B 提高数据库安全性
- ☐ C 提高INSERT 语句的性能
- ☒ D 提高 SELECT 语句的性能

正确答案：D 你的答案：未作答

官方解析：暂无官方题目解析，去讨论区看看吧！

题友讨论(0) ▼

#### 单选题

### 14. 您需要从您的数据库中删除名为 EmployeeView 的视图。应使用哪条语句

- ☐ A DELETE EmployeeView
- ☐ B DELETE VIEW EmployeeView
- ☐ C DROP EmployeeView
- ☒ D DROP VIEW EmployeeView

正确答案：D 你的答案：未作答

官方解析：暂无官方题目解析，去讨论区看看吧！

题友讨论(2) ▼

#### 单选题

### 15. 关于NAT说法不正确的是？

- ☐ A 可实现地址转换
- ☐ B 可实现端口转换
- ☒ C IPv6根本不需要NAT
- ☐ D 可同时实现地址和端口转换

正确答案：C 你的答案：未作答

官方解析：暂无官方题目解析，去讨论区看看吧！

单选题

16. 下列哪些不是IPv6过渡技术

- ☐ A 隧道技术
- ☐ B 地址翻译技术
- ☐ C 双栈技术
- ☒ D 应用识别技术

正确答案: D 你的答案: 未作答

官方解析: 暂无官方题目解析, 去讨论区看看吧!

单选题

17. 下面不属于OSI七层模型的是?

- ☐ A 物理层
- ☒ B 聚合层
- ☐ C 传输层
- ☐ D 应用层

正确答案: B 你的答案: 未作答

官方解析: 暂无官方题目解析, 去讨论区看看吧!

单选题

18. `int a[4][5];` 则`a[1]+3`表示

- ☒ A a数组行下标为1、列下标为3的元素的地址
- ☐ B a数组行下标为1、列下标为3的元素的值
- ☐ C a数组第1行的首地址
- ☐ D a数组第3行的首地址



正确答案: A 你的答案: 未作答

官方解析: 暂无官方题目解析, 去讨论区看看吧!

题友讨论(2) ▼

单选题

19. typedef union{  
    char mark[3];  
    struct{  
        unsigned short rsv:7,  
        type:5,  
        log:1,  
        log\_begin:1,  
        log\_end:1;  
    }flags;  
    unsigned short value;  
}options;  
int main(int argc, char \*argv[])  
{  
    options a;  
    a.value = 1000;  
    printf("%d, %d\n", sizeof(a), a.flags.type);  
    return 0;  
} 请问输出结果是什么?

A 4,29

B 3,7

C 8,29

D 4,7

正确答案: D 你的答案: 未作答

官方解析: 暂无官方题目解析, 去讨论区看看吧!

题友讨论(3) ▼

单选题

20. 以下程序在little-endian架构处理器下的输出是:  
int i;  
char \*pc = &i;  
pc[0] = 1;  
printf("%d\n", i);

A 1

B

16777216

- ☐ C 0
- ☐ D Segmentation Fault

正确答案: A 你的答案: 未作答

官方解析: 暂无官方题目解析, 去讨论区看看吧!

题友讨论(12) ▼

多选题

21. 下面哪一个命令可以关机, 而不重启?

- ☐ A reboot
- ☐ B shutdown -r

☒ C halt

☐ D netstat

正确答案: CD 你的答案: 未作答

官方解析: 暂无官方题目解析, 去讨论区看看吧!

题友讨论(8) ▼

多选题

22. 下列属于Linux开机启动过程的是?

☒ A 运行第一个进程init (进程号永远为1)

☒ B 读取MBR的引导文件 (grub, lilo)

☒ C 引导linux内核

☒ D 进入相应的运行级别

正确答案: ABCD 你的答案: 未作答

官方解析: 暂无官方题目解析, 去讨论区看看吧!

题友讨论(0) ▼

多选题

23. 下列关于inode说法正确的是?

- ☒ A 每一个文件都有对应的inode，里面包含了与该文件有关的一些信息
- ☒ B 特殊文件(比如乱码文件名)可以通过inode的方式删除
- ☐ C inode节点是一个128字节长度的表
- ☐ D 通过inode只能文件读取节点号，获取不到文件相关信息

正确答案: AB 你的答案: 未作答

官方解析: 暂无官方题目解析，去讨论区看看吧!

题友讨论(4) ▼

多选题

24. 关于bash中以下符号的说明哪些是正确的

- ☒ A \$0代表脚本的名称
- ☒ B \$@代表所有位置参数
- ☒ C \$# 代表位置参数的数量
- ☐ D \$11代表第11个位置参数的值

正确答案: ABC 你的答案: 未作答

官方解析: 暂无官方题目解析，去讨论区看看吧!

题友讨论(8) ▼

多选题

25. 关于 DELETE和 TRUNCATE TABLE的说法正确的是

- ☐ A 两者都可以删除指定条目的记录
- ☒ B 前者可以删除指定条目的记录，后者不能
- ☐ C 删除整张表的数据， delete比truncate更高效
- ☒ D 删除整张表的数据， truncate比delete更高效

正确答案: BD 你的答案: 未作答

官方解析: 暂无官方题目解析，去讨论区看看吧!

多选题

26. 下面语句中表示过滤条件的是 vend id=1002或 vend id=1003的是

- ☒ A select \* from products where vend\_id=1002 or vend\_id=1003
- ☒ B select \* from products where vend id in (1002, 1003)
- ☐ C select \* from products where vend id not in (1004, 1005)
- ☐ D select \* from products where vend id=1002 and vend id=1003

正确答案: AB 你的答案: 未作答

官方解析: 暂无官方题目解析, 去讨论区看看吧!

多选题

27. 以下哪些是TCP协议运行时阶段

- ☒ A 连接创建
- ☒ B 数据传送
- ☐ C 数据校验
- ☒ D 连接终止

正确答案: ABD 你的答案: 未作答

官方解析: tcp协议三个阶段 创建 传输 终止

多选题

28. 以下关于超文本传输安全协议说法中正确的说法有

- ☐ A 使用非对称加密算法进行数据传输
- ☐ B 使用对称加密算法进行数据传输
- ☒ C 非对称加密和对称加密都使用了
- ☐ D 其他

正确答案：C 你的答案：未作答

官方解析：证书获取阶段使用非对称加密，数据传输阶段使用对称加密

题友讨论(22) ▼

多选题

29. 以下关于typedef的叙述正确的是

- ☒ A 用typedef可以定义各种类型名,但不能用来定义变量
- ☐ B 用typedef可以增加新类型
- ☒ C 用typedef只是将已存在的类型用一个新的名字来代表
- ☒ D 使用typedef便于程序的通用

正确答案：ACD 你的答案：未作答

官方解析：暂无官方题目解析，去讨论区看看吧！

题友讨论(0) ▼

多选题

30. void alloc\_memory(char \*var, const int size)

```
{
    var = malloc(size);
}
int main()
{
    char *s;
    alloc_memory(s, 10);
    strcpy(s, "abc");
    free(s);

    return 0;
}
```

- ☒ A s未初始化
- ☒ B 内存泄漏
- ☒ C segmentation fault
- ☐ D bus error.

正确答案：ABC 你的答案：未作答

官方解析：暂无官方题目解析，去讨论区看看吧！

## 题友讨论(3) ▼

编程题

9

### 31. 老板发奖金

老板一共需要给某个员工发奖金n元，可以选择一次发1元，也可以选择一次发2元，也可以选择一次发3元。请问老板给这位员工发放完n元奖金共有多少种不同的方法？

数据范围：1 <= n <= 10

你的答案：未作答

官方解析：暂无官方题目解析，去讨论区看看吧！

## 题友讨论(39) ▲

推荐 最新 楼层

在这里输入你的观点吧



秒速一千米 进阶牛

Java

分析：可以这样想，发5元怎么发？

1：先发1块的情况下，剩下4块是不是就和发4块的方法一样了？

2：先发2块的情况下，剩下3块是不是就和发3块的方法一样了？

3：先发3块的情况下，剩下2块是不是就和发2块的方法一样了？

4：先发4块的情况下，剩下1块是不是就和发1块的方法一样了？

5：5块一次性发完，唯一方法

这很递归嘛~

即符合  $f(n) = f(n-1) + f(n-2) + \dots + f(1) + 1$

为便于理解，本人画了张图。

代码和运行结果如下

```
public class GiveMoney {
    public static void main(String[] args) {
        Scanner scanner = new Scanner (System.in);
        System.out.print ("输入要发的奖金:");
        int number = scanner.nextInt ();
        System.out.println ("您有" + f (number) + "种方法发完" + number + "元奖金!!");
    }

    /**
     * 获取 发奖金可用的总方法 的方法
     *
     * @param number 要发的钱数
     * @return 总方法数
     */
    public static int f(Integer number) {
        // 设置递归结束条件
        if (number == 1) {
            return 1;
        }
        // 实现 f(n) = f(n-1) + f(n-2) + ... + f(1) + 1
        int count = 0;
```

```
        for (int i = number - 1; i >= 1; i--) {
            count = f(i) + count;
        }
        return count + 1;
    }
}
```



收起 ^

👍 12    💬 回复    发布于 2022-01-22 22:33



牛客140896412号 回复 这不是梦

只有三种面值：1元、2元、三元

👍 2    💬 回复    发布于 2022-02-15 23:19



offer捏 回复

你这是错的吧

👍 1    💬 回复    发布于 2022-05-18 10:51

展开4条回复 ∨



我要逆天当学霸

C++

```
import java.util.*;
public class Solution {
    /**
     *
     * @param num_money int整型 奖金的总数,单位为元
    ... 展开 ∨
```

👍 5    💬 回复    发布于 2022-01-20 21:32



这不是梦 回复

你好，请问你怎么得到公式：dp[i] = dp[i - 1] + dp[i - 2] + dp[i - 3]? ? ? ?

按你的公式，当i=4的时候，结果是7，应该是符合测试用例（你代码通过了）。但是我一个一个列出来，得到的是8

👍 点赞    💬 回复    发布于 2022-02-10 13:51



这不是梦 回复

哦。我明白了，要看题目。注意：一次性最多只能发3块。。。。。

👍 点赞    💬 回复    发布于 2022-02-18 17:07

展开3条回复 ∨



橘子橘子999

Java

这不就是 青蛙跳吗

👍 4    💬 回复    发布于 2022-02-23 22:26



牛客95393832号

Java

```
int a = 0;
```

```
if( num_money >= 3){
```

... 展开 ∨

👍 2    💬 回复    发布于 2022-01-19 14:43



我叫流弊 回复

4的话明明算出来有8种啊

👍 点赞    💬 回复    发布于 2022-01-22 14:57



牛客342188509号 回复 我叫流弊

我算出4有8种、5有16种

👍 点赞    💬 回复    发布于 2022-01-29 14:18

 这不是梦  青出于蓝 回复


4的时候，我也算出了8.楼主，你能解释一下怎么得到7吗

👍 点赞    💬 回复    发布于 2022-02-10 13:53

 这不是梦  青出于蓝 回复 这不是梦  青出于蓝



如果楼主看见，希望帮忙一下，我这里好纠结。

👍 点赞    💬 回复    发布于 2022-02-10 13:54

 牛客350349617号  小白牛 回复 这不是梦  青出于蓝

题目里说了最多只能发三块钱

👍 点赞    💬 回复    发布于 2022-02-28 13:56

 牛客115221734号  青出于蓝 回复

这个和爬楼梯一样

👍 点赞    💬 回复    发布于 2022-03-18 16:42

收起回复 ^

 TPH-BETTER.  出师牛



Java

以5元为例

先发1元，剩下4元就和一共发4元的方法一样了。



... 展开 ✓

👍 2    💬 回复    发布于 2022-05-09 11:36

 牛客202226209号  飞黄腾达 回复

都写错了操

👍 点赞    💬 回复    发布于 2022-06-28 11:51

 牛客99025299号  小白牛 回复

他这个是对的，已跑代码验证

👍 点赞    💬 回复    发布于 2022-07-13 16:18

 牛客99025299号  小白牛

前端工程师

题解

题解

```
int CalulateMethodCount(int num_money ) {
```

... 展开 ✓

👍 1    💬 回复    发布于 2022-07-13 16:22

 牛客877676053号  青出于蓝


广东外语外贸大学 计算机类

分享一下js的

```
function CalulateMethodCount( num_money ) {  
    // write code her  
    if (num_money == 1) return 1;
```

... 展开 ✓

👍 1    💬 回复    发布于 2022-08-25 18:16 广东

 sssgggjijj  飞黄腾达

门头沟学院 计算机类

```
import java.util.*;
```

```
public class Solution {  
    /**
```

... 展开 ✓





模因抹杀 飞黄腾达

门头沟学院 计算机类

[https://blog.csdn.net/weixin\\_43136158/article/details/108064736](https://blog.csdn.net/weixin_43136158/article/details/108064736) 搞不懂为什么思路是动态规划而不是递归，以及动态转移方程是怎么出来的

点赞 回复 发布于 2022-01-21 22:45



OFFER/OFFER 大橘已定

门头沟学院

简单动规

class Solution

... 展开

点赞 回复 发布于 2022-03-17 18:34

收起

首页

<

1

2

>

尾页

编程题

### 32. 撤销与恢复

撤销/恢复操作具有广泛的用途，比如word文档中输入一个单词，可以点撤销，然后可以再恢复。

编程实现如下功能：从标准输入读取到一个字符串，字符串可包含0个或多个单词，单词以空格或者tab分隔；如果遇到 "undo" 字符串，表示"撤销"操作，前一个字符被撤销掉；如果遇到"redo"字符串，表示恢复刚才撤销掉的字符串。

例如：输入字符串 "hello undo redo world."，对字符串中的 undo 和 redo 处理后，最终输出的结果为 "hello world."

你的答案：未作答

官方解析：暂无官方题目解析，去讨论区看看吧！

题友讨论(32)

推荐

最新

楼层

在这里输入你的观点吧



Pein531 红名大佬

中移研究院 人工智能与智慧运营中心 AI算法工...

先初始化两个栈stack和redo，然后利用双栈求解。遍历词表：

... 展开

26 回复 发布于 2022-01-08 20:08

< 退出答题

【2020】奇安信秋招C/C++方向试卷3

赋个C++ 版

```
#include<iostream>
```

```
#include<queue>
```

```
#include<sstream>
```

```
using namespace std;
```

```
int main(){
```

```
    deque<string> text_queue,undo_queue;
```

```
    string word;
```

```
    while(cin >> word){
```

```
        if(word == "undo"){
```

```
            if(!text_queue.empty()){
```

```
                undo_queue.push_back(text_queue.back());
```

```
                text_queue.pop_back();
```

```
            }
```



```
        }
```

```
    } else if(word == "redo"){
```

```
        if(!undo_queue.empty()){
            text_queue.push_back(undo_queue.back());
            undo_queue.pop_back();
        }
    }
    else{
        text_queue.push_back(word);
        undo_queue = deque<string>();
    }
}

for(int i = 0; i < text_queue.size(); i++){
    cout << text_queue[i] << " ";
}
}
} 收起 ^
```

👍 2    💬 回复    发布于 2022-01-21 22:42


 牛客837553673号  出师牛 回复

也没说撤销后输入新词不能再恢复啊

👍 1    💬 回复    发布于 2022-07-21 15:51

展开3条回复 v



牛客442788909号  飞黄腾达

Java

```
import java.util.*;
public class Main{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        List<String> li = new LinkedList(Arrays.asList(sc.nextLine().split(" ")));
    }
} ... 展开 v
```

👍 4    💬 回复    发布于 2022-01-22 21:09

爱吃葱    回复

Talk are undo is redo cheap. Show undo undo redo redo me the code  
怎么系统答案是: Talk is cheap. Show me the code, 我感觉系统有错啊, 明明先删除了are, 又添回来

👍 1    💬 回复    发布于 2022-03-19 18:12

南风梦西洲    回复    爱吃葱

可能是想简单了, undo后面紧紧跟上redo才能恢复回来

👍 1    💬 回复    发布于 2022-03-25 21:53

Simonscat

中国地质大学(武汉) 测绘类

```
function changeStr( str ) {
    // write code here
    var result = ''
    var splitStr = str.split(' ')
} ... 展开 v
```

👍 2    💬 回复    发布于 2022-02-23 19:30

橘子橘子999    回复

我和你差不多用的split

👍 点赞    💬 回复    发布于 2022-02-23 22:28

小居~    回复

```
function changeStr(str) {
    let str2 = str.trim()
    let undoArr = [] // 把被撤销的字符串放里面来
    let redoArr = [] // 恢复刚刚撤销的字符串, 放里面来
    let strArr = str.split(' ') // 有空格的地方截止, 然后就是一个元素, 放到数组里面 ['hello','undo','redo','world'] ... 展开 v
```

👍 点赞    💬 回复    发布于 2022-03-03 22:26

TPH-BETTER.  
Java

先初始化两个栈stack和redo，然后利用双栈求解。遍历词表：

... 展开

2 回复 发布于 2022-05-09 15:05

秒速一千米  
Java



请指教

```
public class UndoRedo {  
    public static void main(String[] args) throws InterruptedException {
```

... 展开

1 回复 发布于 2022-01-22 21:07

HelloWorld程序员 回复

棒!

点赞 回复 发布于 2022-02-10 10:17

lbw\_993  
后端

```
import java.util.*;  
  
public class Main {  
    public static void main(String[] args) {
```

... 展开

1 回复 发布于 2022-03-30 19:07

嘻哈王  
Java

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

... 展开

点赞 回复 发布于 2022-02-13 16:55

OFFER/OFFER  
门头沟学院

```
#include <bits/stdc++.h>  
using namespace std;  
int main()  
{
```

... 展开

点赞 回复 发布于 2022-03-17 18:32

KingMaker  
华南农业大学 计算机类

```
let readline=require('readline')  
  
const r=readline.createInterface({  
    input:process.stdin,
```

... 展开

点赞 回复 发布于 2022-03-21 22:04

offer砸我。  
门头沟学院 计算机类

```
package main  
import(  
    "os"  
    "bufio"
```

... 展开

点赞 回复 发布于 2022-03-22 09:08

收起

首页

<

1

2

3

>

尾页

选择试卷，继续练习