

N 诺考研系列

计算机考研机试攻略

满分篇



N 诺课程教研团队

2019. 12. 01 更新

写在前面的话

相信各位同学都知道，N 诺是一个大佬云集的平台。N 诺每周都会定期举办比赛，让同学们在日复一日的枯燥生活中找到一种乐趣。

本书是 N 诺专为计算机考研的同学精心准备的一本**神书**，为什么说它是一本神书，它到底神在什么地方？

- 1、这本书是 N 诺邀请众多 **CSP 大佬**、**ACM 大佬**、**BAT 专业大佬**以及往年**机试高分大佬**共同修订而成。
- 2、这本书与市面上的书都不一样，这是一本专门针对**计算机考研机试**而精心编写而成的书籍。
- 3、这本书上的**例题讲解**以及**习题**都可以在 N 诺上找到进行练习，并且每道题都有大佬们发布的**题解**可以学习。
- 4、N 诺有自己的官方群，群里大佬遍地走，遇到问题在群里可以随时提问。群里**神仙**很多，不怕没有问题，就怕问题太简单。

相信同学们一定听过各种各样的 OJ 平台，但是那些平台不是为了计算机考研而准备的，而 N 诺是**唯一**一个纯粹为计算机考研而准备的学习平台。

相信每一个同学在学习本书之前，都觉得机试是一个很头疼的问题，机试有可能**速成**吗？我可以一周就**变得很强**吗？我们可以在短短两三周之内的学习就能拿到**机试高分**吗？

别人可不可以我不知道，但是在 N 诺这里，你就可以。

虽然考研机试题目千千万，但是你可以“一招鲜，吃遍天”。

管它乱七八糟的排序，我就一个 sort 你能拿我怎么办？

管它查来查去的问题，我会 map 我怕谁？

管它飘来飘去的规律，我有 OEIS 神器坐着看你秀！

管它眼观缭乱的算法，我有 N 诺万能算法模板怕过谁？

本书虽然不能让你变成一个算法高手，但是本书可以让你快速变成一个机试高手，这也是本书最重要的特点，以解决同学们的实际需求为目的。你可以不会算法，你可以不必弄懂算法的原理，你只要学会本书教给你的各种技巧，就足以应对 99.9% 的情况。

N 诺的课程教研团队全是由大佬组成，每个人都做过几千道编程题目，做编程题的经验十分丰富，我们分析近百所院校的历年机试真题，从中发现了各个学校的机试真题都有相同的规律，万变不离其中，于是本书也就诞生了。

计算机考研机试攻略交流群请扫描下方二维码或直接搜索群号：960036920



群名称：N诺 - 计算机机试交流群
群号：960036920

本书配套了课后习题视频讲解，请加入计算机考研机试攻略交流群查看群公告。

关于 N 诺

N 诺是全国最大的计算机考研在线学习平台。N 诺致力于为同学们提供一个良好的网络学习环境，一个与大佬们随时随地交流的舒适空间。如果你不知道 N 诺，那么你已经输在起跑线上了，因为 N 诺是 - 计算机学习考研必备神器。

N 诺收集并整理了**计算机考研的前世今生**，帮助你了解考研的点点滴滴。

<http://www.noobdream.com/post/585/>

在 N 诺，你可以查询各个院校的**考研信息**，包括分数线、录取人数、考试大纲、导师信息、经验交流等信息，应有尽有。

<http://www.noobdream.com/schoollist/>

在 N 诺，你可以尽情的刷各个科目的**题库**，还可以将题目加入**错题本**方便以后复习，也可以写下自己的**学习笔记**，记录考研过程中跌宕起伏。

<http://www.noobdream.com/Practice/index/>

在 N 诺，你可以在**讨论区**里发表你的问题或感想，与全国几百万考研 er 分享你的喜怒哀乐。

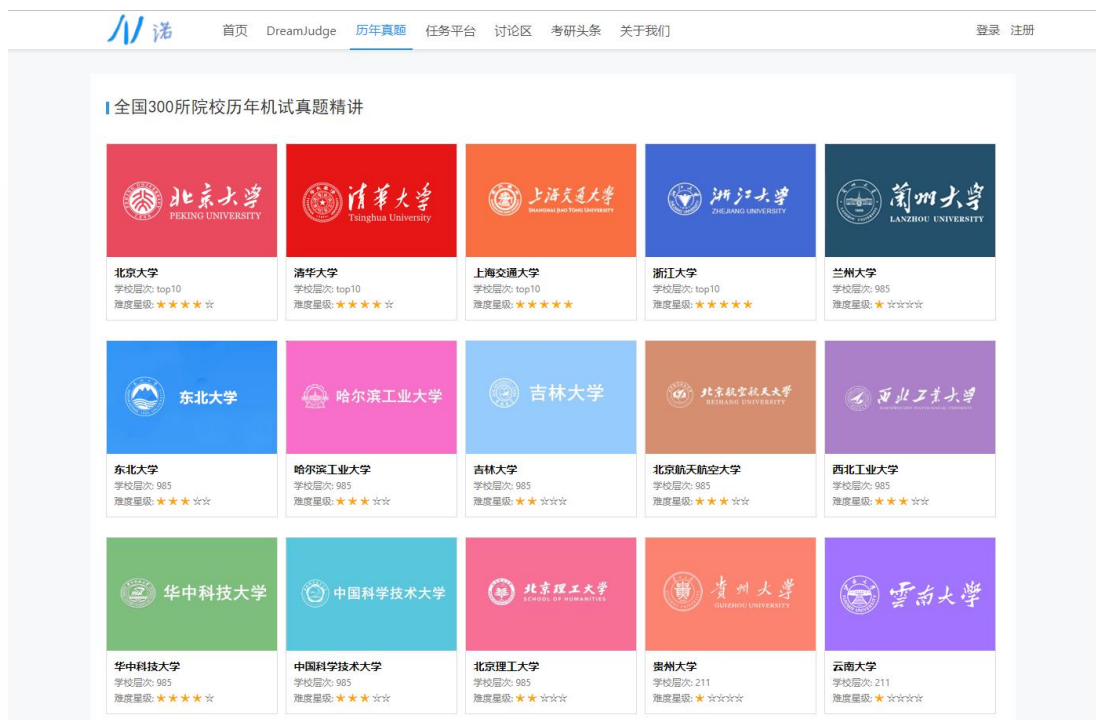
<http://www.noobdream.com/forum/0/>

在 N 诺，你可以将你不用**的书籍或资料**放到**二手交易市场**，既能帮助他人，还能为自己省下一笔当初买书买资料的费用。

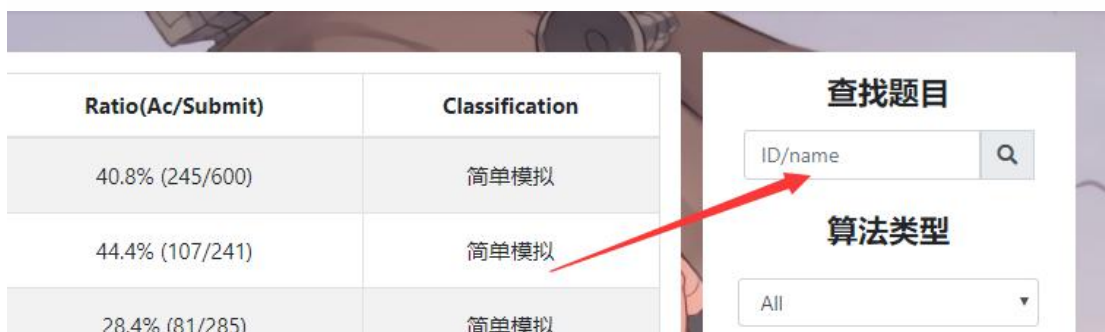
<http://www.noobdream.com/Task/tasklist/>

如何使用本书？

访问 N 诺平台（www.noobdream.com）的历年真题，即可查询到全国各个学校的历年机试真题。



如下图输入课后习题的编号即可搜索到题目



小技巧：N 诺是可以随意更换皮肤的哟，在右上角头像旁边。

第一章 技巧之巅

建议大家在阅读本书之前先阅读本书的姊妹篇：计算机考研机试攻略 - 高分篇。

这本书是高分篇的加强版，适合学习完高分篇的同学继续学习，拿到机试满分。

本章是N诺计算机考研机试攻略满分篇的第一章，本章我们带领大家领略技巧的极致魅力。本章包括：输入输出加速外挂、空间换时间技巧、位运算技巧、考试最佳策略、对数器技巧等内容。可以帮助读者迅速掌握一些使用的考试技巧，完成更高难度的挑战。



1.1 输入输出加速外挂

有的时候题目的输入数据量比较大，比如要输入 10W 和数字进行排序。这个时候，如果你直接使用 C++ 的 `cin` 和 `cout` 函数进行输入输出，有很大的概率会超出题目的时间限制。

在这种情况下，需要的优化的就不再是你的算法过程，而是你的读写数据的速度优化。如果使用 `cin` 和 `cout` 函数进行输入输出，我们建议你在 `main()` 里首先写入下面两行代码。

C++ cin/cout 加速

```
1. ios_base::sync_with_stdio(0);
2. cin.tie(0); cout.tie(0);
```

如果题目的输入量巨大，比如要输入 100W 个数字，这个时候我们建议你使用 C 语言的 `scanf` 和 `printf` 语句进行输入输出。

如果在这种情况下还超时，那么你就需要用到下面的输入输出加速外挂了。

```
1. //适用于正负整数
2. template <class T>
3. inline bool scan_d(T &ret) {
4.     char c; int sgn;
5.     if(c=getchar(),c==EOF) return 0; //EOF
6.     while(c!='-'&&(c<'0' || c>'9')) c=getchar();
7.     sgn=(c=='-')?-1:1;
8.     ret=(c=='-')?0:(c-'0');
9.     while(c=getchar(),c>='0'&&c<='9') ret=ret*10+(c-'0');
10.    ret*=sgn;
11.    return 1;
12. }
13. inline void out(int x) {
14.     if(x>9) out(x/10);
15.     putchar(x%10+'0');
16. }
```

加速外挂原理:

getchar 的速度 **快于** scanf 的速度

速度比较

getchar > scanf > cin

Putchar > printf > cout

N 诺
noobdream.com

1.2 调试技巧

调试是我们在编写程序时不得不经历的过程, 我们这一节来讲解如何快速定位错误的调试技巧。

首先, 编译错误不在我们的讨论范围之内, 语法上的错误问题相信学过高分篇的同学都不会再问了。

其实, 断点调试也不在我们的讨论范围之内。

这里讲一下我们为什么不建议大家使用断点调试, 断点调试基本上是每个开发者都会的调试技巧。

但是我们的机试有一些特殊的情况。

- 1、机试的代码往往很短, 几行到几十行不等。
- 2、比赛中争分夺秒, 我们对调试时间要求更为迫切。
- 3、我们的错误往往是由于代码细节没考虑周全导致的。

所以, 断点调试更适于项目代码中且对时间的迫切度没有那么高的情况。

接下来, 给大家介绍一种超级棒的调试技巧, 当你熟练掌握它以后你会深深的迷恋上它。

输出调试

顾名思义，就是通过输出的方式定位我们的错误所在。

大部分没有足够调试经验的同学使用输出调试的时候，不知道应该如何使用输出调试。

如果从前往后逐条语句输出调试去排查错误，那么很容易要找很久。

反之从后往前逐条语句输出调试去排查错误，那么也很容易要找很久。

不知道同学们是不是发现了点什么，对，没错，上面两种方法是不是就是顺序查找的方法。

那么，与此对应的就应该是二分查找的方法。

使用二分查找的思想来调试定位错误，可以更快，更节约时间。

noobdream.com

特别注意：输出调试完成之后提交代码之前一定要删除或注释掉调试信息。

1.3 位运算技巧

位运算，相信所有的同学都有所了解。

那么关于位运算有哪些特殊的技巧呢？

速度比较

取模时间 > 四则运算时间 > 位运算时间

所以对于一个语句

```
1. if (a % 2 == 1) {  
2.     a /= 2;  
3. }
```

我们可以优化成为

```
1. if (a & 1 == 1) {  
2.     a >>= 1;  
3. }
```

异或运算的特殊性

异或同一个数 2 次或者偶数次，那么本身的值不变。

例如：

$$a \oplus b \oplus b = a$$

$$x \oplus y \oplus y \oplus y \oplus y = x$$

接下来我们来看一下它的应用。

缺页问题

题目描述：

小明有一本很喜欢的漫画书，但是由于他上课的时候偷看这本漫画书被老师发现，于是老师将

小明的漫画书撕了扔进垃圾桶。

小明放学后从垃圾桶里将漫画书捡了回来，却发现漫画书少了一页。小明的漫画书一共有 N 页，由于现在页码顺序错乱，小明也不知道是少了哪一页，聪明的你能告诉小明缺失的是哪一页吗？

输入描述：

第一行输入一个整数 N ($N < 1000000$)，表示小明书的总页数。

第二行输入 $N-1$ 个数整数，代表小明找到的页码。

输出描述：

请输出小明缺失的页码在单独的一行。

请注意：本题的空间很小，要求你尽量不使用额外的空间来解决。

输入样例#：

6

6 2 1 5 3

输出样例#：

4

题目来源：

DreamJudge 1506

题目解析：由于本题要求我们以尽量小的空间来解决问题，所以我们不能够使用数组来存储每一个数。那么我们应该怎么办呢？这个时候可以想到异或运算的特殊技巧，同一个数异或两次那么就会消除，如果我们提前将 1 到 N 的所有数字进行异或处理，然后再去异或输入的 $N-1$ 个数，那么答案就是缺失的那个数。

参考代码

```
1. #include<bits/stdc++.h>
2. using namespace std;
3.
4. int main(){
```

```
5.     int n, x;
6.     scanf("%d", &n);
7.     int sum = 0;
8.     for (int i = 1; i <= n; i++) {
9.         a[i] = i;
10.        sum ^= i;
11.    }
12.    for (int i = 1; i < n; i++) {
13.        scanf("%d", &x);
14.        sum ^= x;
15.    }
16.    printf("%d\n", sum);
17.    return 0;
18. }
```

题目练习

DreamJudge 1118 将军的书

noobdream.com

1.4 考试最佳策略

很多上机考试经验不够丰富的同学，往往很难发挥出自己应有的水平。

下面我们来讲一下如何应对一场考研的上机考试。

提前准备

- 1、算法模板一定要提前准备好，不管是单独打印还是记录在书上。
- 2、算法模板一定要验证它的正确性，在验证的过程中也知道了该如何使用。
- 3、手机调成静音模式，最好再开一个飞行模式，不要上交以备不时之需。

开考前

- 1、验证鼠标键盘是否可用，如果不可用及时向老师反应，更换电脑。
- 2、验证 IDE（如：codeblocks）是否可用，如果不可用及时向老师反应，更换电脑。
- 3、提前将头文件和主函数框架写好，最好创建两个文件，代码复制一份就行。因为在做题过程中遇到卡题情况，可以及时切换到另一道题上继续写。

考试中

- 1、机试中的题目难度不是从简单到难，难度是随机的，所以刚开始一定要将所有题目都看一遍。
- 2、找到你认为最简的那道题开始做，记住，一定要从最简单的题目开始做。
- 3、考试过程中要注意看排行榜，通过人数最多的题目一般都是最简单的题目。
- 4、注意：要看通过人数的多少来判断难易程度，而不是第一个人通过的时间来判断。
- 5、如果你的水平强，可以选择先做那种代码不长但是需要算法思维的题，快速解出来将排行榜带偏，给其他人一种这题最简单的感觉，然后让他们死磕这题，成功让竞争对手翻车。
- 6、之所以要赤裸裸的将上一条写出来，就是为了告诫那些头铁的同学，不要在一棵树上吊死，都走复试这一步了，还这么死脑筋基本没得救了，
- 7、考试的时候遇到规律题，千万不要忘记 OEIS 这个神器，特别是周围上厕所的同学开始变多。
- 8、当无法通过一个题的时候，先看看有没有其他能做的题，如果也没有其他题能做了。

这个时候你就可以使用很多特殊的办法，首先，你一定要相信，机试的数据一定不够强。一般情况下，机试的判题数据都会找学生帮忙生成，往往强度就不会很高，这个要看学生的尽责程度。如果是老师自己构造的数据，你睡着都会笑醒。

9、不要被题目的数据范围吓到，有可能后台都是小数据，没有更好的解法的时候一定要试试暴力。

10、我们往往被卡都是因为算法不够优秀导致超时，一方面我们可以强行优化输入输出加速来看看能不能水过去。另一方面可以采用小数据暴力，大数据随机的思想来解决问题。

小数据暴力、大数据随机

原理：由于大数据出多组容易导致判题很慢，所以往往不会有很多组。另外对于特殊的数据可能需要手动构造，大数据构造起来麻烦，还要自己构思生成数据的代码。所以一般都是用小数据来验证算法正确性，再加上两组大数据组验证算法复杂度。

举例说明

背包问题

我们背包问题一般使用动态规划来解，但是你不会怎么办？

那么我们就可以小数据暴力搜索，大数据直接贪心。对于数据不够强的题就能水过去。

更简单的例子

给你很多个数，要你从中找出最大的数。

你需要从第一个找到最后一个才能判断出最大的数是哪个吗？

如果是这样几个数：5 2 7 1 6 3

我们是不是只需要找到一半就能找到最大那个数了

如果是这样几个数：8 2 7 1 6 3

我们是不是只需要找到第一个就能找到最大那个数了

一般情况下，判题数据就像上面那样，你可以通过不严谨的代码通过题目。

主要是领悟思想，不同的题都有不同策略。

1.5 预处理与打表技巧

预处理

预处理是指将答案提前处理好然后再进行查询的方法。

什么时候会用到预处理？

查询量特别大的时候。

例题

我们要查询斐波那契数列的第 n 项, n ($n < 10000$)。

并且我们要查询 10 万次。

错误的超时解法

```
1. #include<bits/stdc++.h>
2. using namespace std;
3.
4. int f[10005];
5. int main(){
6.     int t;
7.     scanf("%d", &t); //查询次数
8.     while (t--) {
9.         int n;
10.        scanf("%d", &n); //查询第 k 项
11.        f[1] = 1;
12.        f[2] = 1;
13.        for (int i = 3; i <= n; i++) {
14.            f[i] = f[i-1] + f[i-2];
15.        }
16.        printf("%d\n", f[n]);
17.    }
18.    return 0;
19. }
```


超时分析：每一次查询，我们都递推了一遍斐波那契数列，如果每一次查询的都是最后一项。那么最坏情况就是：100000*10000，必然是会超时的。

正确的预处理解法

```
1. #include<bits/stdc++.h>
2. using namespace std;
3.
4. int f[10005];
5. int main(){
6.     f[1] = 1;
7.     f[2] = 1;
8.     for (int i = 3; i <= 10000; i++) {
9.         f[i] = f[i-1] + f[i-2];
10.    }
11.    int t;
12.    scanf("%d", &t); // 查询次数
13.    while (t--) {
14.        int n;
15.        scanf("%d", &n); // 查询第 k 项
16.        printf("%d\n", f[n]);
17.    }
18.    return 0;
19. }
```

可以发现两段代码惊人的相似，有什么区别呢？

区别在于下面的代码只会递推一遍斐波那契数列，然后将所有的答案都存储于 f 数组中。

一般打表技巧

打表是指我们提前使用暴力的方法, 将答案记录下来写到纸上或代码中, 然后再根据题目的实际需求去直接输出提前记录好的答案。

例题

求一个数 x 的 100000000 次方模 2333 的值是多少。

x 的范围是 ($1 \leq x \leq 10$)

解析: 这个时候我们如果不会二分快速幂的话, 可以直接用一个暴力的代码将 1 到 10 的答案分别求出来, 可能要等上几分钟, 然后直接判断输入的值对应输出结果即可。

打表找规律

在一些可能存在规律的问题中, 我们可以通过暴力打表求出前几十项的值, 发现其中的规律, 从而帮助我们解出该题。

例题

输入一个整数 a ($a < 10^9$), b 可以取任意值, 求 $a\%b$ 的可能产生多少种不同的结果。

解析: 乍一看, 我们一脸懵逼。这个时候你只要去打表分析, 首先 b 大于 a 就重复了, 那么我们只需要对于任意一个 a , 去枚举 b 的值, 看 $a\%b$ 会产生多少种不同的结果即可。通过打表分析, 你会发现: 当 $a = 1$ 时, 结果 $ans = 1$, 当 a 为偶数时, $ans = a / 2 + 1$, 当 a 为奇数时, $ans = a / 2 + 2$ 。

题目练习

DreamJudge 1488 数字填充

第二章 满分之路 上

高分篇的内容已经将机试可能涉及到的题型及考点都一一列举出来, 并给大家提供了练习题目, 帮助大家巩固基础加深理解。在大多数院校的机试中, 学会高分篇的内容已经足够考到 90 分以上的成绩, 题目不难的情况下甚至能拿到 100 分。在少数院校难度较大的院校也足够拿到 80 分左右的分数, 如果你基础比较薄弱, 我们建议你掌握高分篇的内容就可以了。本书满分之路的内容不是常考点, 只是有一定可能会考到, 我们建议基础好的同学继续学习增加拿满分的把握。

距离满分只有一步之遥, 你! 愿意放弃吗?



2.1 区间问题

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



2.2 动态规划的优化技巧

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



2.3 尺取法

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



2.4 博弈类问题

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



2.5 最长路径问题

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



2.6 二分答案技巧

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



第三章 满分之路 中

高分篇的内容已经将机试可能涉及到的题型及考点都一一列举出来, 并给大家提供了练习题目, 帮助大家巩固基础加深理解。在大多数院校的机试中, 学会高分篇的内容已经足够考到 90 分以上的成绩, 题目不难的情况下甚至能拿到 100 分。在少数院校难度较大的院校也足够拿到 80 分左右的分数, 如果你基础比较薄弱, 我们建议你掌握高分篇的内容就可以了。本书满分之路的内容不是常考点, 只是有一定可能会考到, 我们建议基础好的同学继续学习增加拿满分的把握。

距离满分只有一步之遥, 你! 愿意放弃吗?



3.1 静态区间查询类问题

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



3.2 动态区间查询类问题

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



3.3 线段树的各种应用技巧

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



3.4 平衡二叉树的问题

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



3.5 字符串匹配类问题

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



3.6 图的连通性问题

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



3.7 图的匹配问题

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



第四章 满分之路 下

高分篇的内容已经将机试可能涉及到的题型及考点都一一列举出来, 并给大家提供了练习题目, 帮助大家巩固基础加深理解。在大多数院校的机试中, 学会高分篇的内容已经足够考到 90 分以上的成绩, 题目不难的情况下甚至能拿到 100 分。在少数院校难度较大的院校也足够拿到 80 分左右的分数, 如果你基础比较薄弱, 我们建议你掌握高分篇的内容就可以了。本书满分之路的内容不是常考点, 只是有一定可能会考到, 我们建议基础好的同学继续学习增加拿满分的把握。

距离满分只有一步之遥, 你! 愿意放弃吗?



4.1 逆元

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



4.2 组合数取模类问题

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



4.3 欧拉降幂

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



4.4 矩阵快速幂

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



4.5 数位类型 DP

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



4.6 树上的 DP

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



4.7 带状态压缩的搜索

试读版，已删减，请登录 N 诺官方平台兑换中心进行兑换。

N 诺 官方群：612180120



完结撒花

当你看到这里, 说明你已经读完了本书所有的内容。恭喜你, 学完计算机考研机试攻略 - 满分篇的全部内容, 我们相信你一定会在机试中取得非常不错的成绩。

如果本书对你有所帮助, 希望你能在复试之后将还能记得的机试题目发表在 N 诺的交流区里, 当然也可以直接在 N 诺官方群或机试群里联系管理员。N 诺会根据你提供题目描述进行数据还原, 继续帮助下一届学弟学妹, 让他们可以做到最新的真题, 少走一些弯路, 并且我们会将你的名字或 N 诺 ID 放在题目的后面进行特别鸣谢。

最后, 我们不仅希望你能在机试中取得满分的成绩, 也希望你能如愿以偿的考上心目中理想的院校, **加油! Go!Go!Go!**

一定要做的说明: N 诺出版的考研系列书籍都将以**电子版**的形式进行发布更新, 需要纸质版的同学自行打印学习即可。

N 诺考研系列书籍为什么只发布电子版不发布纸质版？

原因一：发布纸质版需要提前很久将书籍整理成册，时间太赶容易敷衍了事，我们相信慢工出细活。

原因二：纸质版一经印刷，便无法修改，就算发现问题或者想对某些内容进行优化也没办法。而电子版可以随时勘误进行修改，灵光一现的时候还能对前面写的不够好的地方进行优化。

原因三：电子版少了中间商，可以给同学们节约更多的费用。纸质版的话出版社、印刷商都要从中获取利润，最终羊毛出在羊身上。



如何获取 N 诺考研系列书籍？

noobdream.com

访问 N 诺平台（www.noobdream.com）的兑换中心即可兑换或购买各种你想要的书籍或资料。

另外，本书会不断的进行更新，所以需要最新版的同学，请去官网**兑换中心**进行兑换，只要一次兑换，后续版本更新都可看到，不用重复兑换。

考研机试冲刺八套卷

在本书的最后告诉所有同学一个好消息, 由 N 诺出版的考研机试冲刺八套卷将**免费赠送**给所有考研的同学。

虽然本书已经将各种考点和考法都在书中做了介绍, 但是想到很多同学都没有实际机试的经验, 对本书的介绍的各类做题技巧可能没有适应, 所以需要几场模拟考试来将各种技巧**融会贯通**, 在正式的上机考试中才能游刃有余。

N 诺邀请了众多大佬共同出这八套卷的题, 押中原题的话不要惊讶, 这只是常规操作。毕竟这其中曾经把机试题全部押中的大佬, **押题**, 绝对是专业的。当然, 这八套冲刺卷目的不仅仅是为了押题, 更是为了帮助同学们**快速变强**, 题目难度适中(准确说应该是较为简单), 适合所有学校的同学。

最最最重要的是, 在每一场比赛结束之后的 10 分钟以内, 我们会马上在 B 站上直播讲解这套冲刺卷的每一道题的题解(**全程讲解答疑免费**), 以及可以使用什么样的技巧**快速通过**这类题, 毕竟有很多实用的技巧在书上不太好讲述, 实战过程中大家也更容易理解。

直播题解传送门: <https://live.bilibili.com/3513962> (可以提前关注哦)

还要一点很重要, 同学们看了直播题解懂了怎么做之后, 一定要去**补题**, 不补题的话, 你绝对会后悔的。对于绝大部分初学者来说, 口头 AC 和实际 AC 中间差了十万八千里。

温馨提示：这八套冲刺卷是非常重要的，就算你是一个什么都不会的小白，只要认真做了这八套卷，赛后把题都补了，绝对会拿到机试高分。如果你认真做了却没能拿到高分，我们一起祝N诺明年倒闭。

21 考研的同学也可以来做这八套卷，今年提前做了，明年就轻松了。

如何报名？

比赛网址：<http://www.noobdream.com/DreamJudge/Contest/match/>

点击报名即可参加，没有提前报名的话，比赛过程中也可报名。

比赛时长：3 个小时

题目数量：6 个题

时间安排

考研机试冲刺八套卷（第一套）：12 月 28 日 晚上 6 点 — 9 点

考研机试冲刺八套卷（第二套）：1 月 4 日 晚上 6 点 — 9 点

考研机试冲刺八套卷（第三套）：1 月 11 日 晚上 6 点 — 9 点

考研机试冲刺八套卷（第四套）：1 月 18 日 晚上 6 点 — 9 点

考研机试冲刺八套卷（第五套）：1 月 25 日 晚上 6 点 — 9 点

考研机试冲刺八套卷（第六套）：2 月 1 日 晚上 6 点 — 9 点

考研机试冲刺八套卷（第七套）：2 月 8 日 晚上 6 点 — 9 点

考研机试冲刺八套卷（第八套）：2 月 15 日 晚上 6 点 — 9 点

即每周六晚 6 点开始，9 点结束，9 点 10 分开始直播讲解每道题的解法。

最后，每一场比赛开始前，我们会在N诺官方群(612180120)和N诺考研机试交流群(960036920)提醒各位同学参加比赛，赛后也会将题解和标程上传到群文件，大家有不会的问题也可以在群里进行交流探讨。

N 诺网校

在本书的最后给大家安利一下 N 诺网校，N 诺网校的**性价比**可以说是无法用言语形容的。

- 1、N 诺网校没有营销环节，可以为大家节约 **20-30%**的费用。
- 2、N 诺网校没有广告支出，可以为大家节约 **30-50%**的费用。
- 3、N 诺网校不抽取上课老师的分成，可以为大家节约 **50-60%**的费用。
- 4、N 诺网校有渠道给大家找到专业大佬来上课，水平有保证。
- 5、由于网校这一块是亏本运营，所以没有固定开班时间，可以**预报名**，报名人数达到最低开课人数即开始上课，**组团报名**不仅有优惠，而且可以更快开课。



该截图展示了 N 诺网校的课程列表。页面顶部有导航栏，包含“首页”、“DreamJudge”、“历年真题”、“任务平台”、“讨论区”、“考研头条”和“兑换中心”。右侧有“登录”和“注册”按钮。课程列表标题为“N诺网校课程”，下方列出了五门课程：

- 零基础入门课程**：学完本次课程，可以基本掌握C/C++语言技能，能完成大部分基础编程题目和一些简单的算法题目。长期课程，编程入门。¥1，开始时间 2019年8月30日 00:52，报名咨询人数: 245。
- 算法竞赛基础课程**：学完本次课程，可以基本掌握各类算法竞赛基础算法知识，可以继续进阶算法竞赛提高课程的学习。长期课程，算法基础。¥1，开始时间 2019年8月30日 01:30，报名咨询人数: 157。
- 算法竞赛提高课程**：学完本次课程，可以掌握一些进阶算法、数据结构和数学知识，足以应对各类算法竞赛，并获得不错的成绩。长期课程，算法提高。¥1，开始时间 2019年8月30日 01:31，报名咨询人数: 84。
- 考研机试零基础班**：学完本次课程，你可以学会写大部分简单的代码。7天限时，零基础。¥1，开始时间 2019年11月21日 11:03，报名咨询人数: 135。
- 考研机试提高班**：学完本次课程，你可以学会写大部分简单的算法。7天限时，提高。¥1，开始时间 2019年11月21日 11:05，报名咨询人数: 145。

N 诺网校的课程分为两种模式

- 1、小班课程
- 2、一对一个性化辅导

小班课程除了上课老师以外，还会有 2-3 名助教进行日常答疑，价格也非常实惠，一门课 30 课时左右，一门课仅需 150-300 元，并且附赠培训辅导书以及教学视频。

一对一个性化辅导适合经济宽裕点的同学，虽然没有动辄上万的费用，但是 100 元/小时差不多是最低价格了，低于这个价格几乎不可能有大佬愿意接单。

最关键的是：N 诺网校可以上一次课交一次课的费用，这是别的地方不敢想象的。

N 诺网校的科目

- 1、C 语言
- 2、数据结构
- 3、操作系统
- 4、计算机组成原理
- 5、计算机网络
- 6、数据库
- 7、算法分析
- 8、考研机试

如何报名？

在 N 诺官方群联系管理员或者在 N 诺网校的对应科目中点击报名意向咨询即可。