

再版前言

历时n周，终于完成。

本书前面为简单的c++基础，后面为简单的算法基础，打星号的题目表示为算法部分，这部分希望一同做开发的同学必备。

本书目录为学习顺序，从level1-5，具体题目在正文，Example是例题，建议学习C++相应内容后检验下自己基础是否掌握，或者可以找我给你当做例题讲解。Problem一定要自己完成，任何高手都是在实践中成长。另外没有题目的重要知识，请自学！

本文Level6的部分是属于算法部分，建议同学们去baidu找相关解释自学，或者几人讨论完成。注意，不要百度源代码或者直接阅读同学源代码就自认为学会了！！原因自己清楚。

正文题目完成后，可以独立上机调试，或者找已经学会相关章节的同学学习、讨论。对于本书写代码空间不够的题目，自己想办法解决——若想从本书学走精华的，不会为这个难倒的，反而如果你拖拖拉拉的，那么这份一个月可以完成的教程，拖多久则都有可能。Linyh

CSDN

Level 1.1 入门

P1001.Example（基础）输入两个数，求出它们的和并输出。



样例输入：1 2

样例输入：3

P1002.Problem（基础）输入立方体的长宽高，求出它的

样例输入：1 2 3

样例输入：6 22



Level 1.2 数据类型

P1003.Example（数据类型入门）输入三角形的底和高 a, h ， $a, h \in (\mathbb{N}^*)$ ，求面积并输出

样例输入：1 1

样例输入：0.50000

P1004.Example（常量）输入圆的半径，求周长。

样例输入：0.5

样例输入：3.14159

P1005.Problem（char，ASCII）读入一个字符，输出其ASCII码

样例输入：a

样例输入：97

Level 1.3 标准库

P1006.Example（`#include` 标准库函数）输入三角形的三边，求面积与周长

样例输入：3 4 5

样例输入：6.00000 12

P1007.Problem（`<cmath>`）读入一个数 x ，求 e 的 x 的绝对值次幂。用`exp()` `abs()`

样例输入：-1

样例输出：2.71828

Level 2 条件

P1008.Example（逻辑运算符）输入一个年份，判断是不是闰年

样例输入1：2008

样例输入1：Yes.

样例输入2：2100

样例输入2：No.

P1009.Example（`switch`）输入成绩（分数），输出等第， $>90A, 80-90B, 70-80C, 60-70D, <60E$

样例输入：60

样例输入：D

P1010.Example（`if`语句入门） $ax^2+bx+c=0$

输入一元二次方程的 a, b, c （均为整数），求出 x_1, x_2 。无解则输出"No solution."

样例输入1：1 -3 200

样例输入1：No solution.

样例输入2：1 -3 2

样例输入2： $x_1=1.00000$ $x_2=2.00000$

P1011.Example（`?:`）输入三个数，输出最大的数。注意，不要用`if`而是用`?:`表达式

样例输入：3 4 5

样例输入：5

P1012.Problem (if) page 86 title 10

$y=x(x<1)$

$y=2x-1(1\leq x\leq 10)$

$y=3x-11(x>10)$

样例输入：4

样例输出：7

P1013.Problem (if) page 86 title 12 输入一个整数n ($n<1000$)，将这个数倒序输出

样例输入：123

样例输出：321

P1014.Problem (if) page 86 title 12

收电费，电费的单价，小于等于50kwh的部分0.5元，50-100 0.7，100以上1元每kwh,，输出电费总价。

样例输入：51

样例输出：25.7

P1015.Problem (if) 输入在一点处的一个半径为r的圆锥，这个圆锥高为h，有输入一点坐标(x, y)，求此点的高度（除了圆锥以外各点高度均为0）。

样例输入1：5 10 3 4

样例输出1：0

样例输入2：10 10 3 4

样例输出2：5.00000

P1016.Problem (?:)

$y=x-1(x<1\text{或者}x>10)$

$y=x+2(1\leq x\leq 10)$

样例输入：3.5

样例输出：5.5

P1017.Problem (switch) 输入年份以及月份，输出这个月有几天

样例输入：2008 2

样例输出：29

P1018.Problem (逻辑运算符) 输入成绩a及素质分b，若a,b都 ≥ 90 ，则输出great。若a,b有一个 ≥ 60 ，则输出so-so。否则，则输出bad

样例输入：60 20

样例输出：so-so

Level 3 循环

Break continue没有题目

P1019.Example (for循环语句) 输入一个数 n , $n \in \mathbf{N}^*$, 如果是素数(质数), 则输出yes。

样例输入1: 3

样例输入1: Yes.

样例输入2: 9

样例输入2: No.

P1020.Example (while循环) 输入 n ($0 < n < 10$), 求使 $1/1+1/2+1/3+\dots+1/k > n$ 成立的最小正整数 k

样例输入: 2

样例输入: 4

P1021.Example (do...while循环语句) 输入很多数, 直到输入-1时停止。

样例输入: 1 3 5 7 8 23 3 5 67 -1

P1022.Example (for循环的嵌套) 输入 n , 输出如下图形

样例输入: 5

样例输入:

**

*

P1023.Problem (for&&for) page 86 title17

$S_n = a + aa + aaa + aaaa + \dots + a(n\text{个})$, a, n 输入的数 $a, n \in \mathbf{N}^*$ 且 $a \in [0, 9], n \in [2, 9]$ 。

比如 $a=8$ 时, $S_4 = 8 + 88 + 888 + 8888 = 9872$

样例输入: 8 4

样例输出: 9872

P1024.Problem (for&&for) page 86 title18

求 $1! + 2! + 3! + 4! + \dots + n!$ 的值并输出。 $n \in [5, 11]$ 。

样例输入: 5

样例输出: 153

P1025.Problem (for&&for) $n=3$ 时输出如下图形

*

*

样例输入: 2

样例输出:

*

*

P1026.Problem (for&&for) page 87 title21

$2/1, 3/2, 5/3, 8/5, 13/8, 21/13, \dots$ 求此数列的前 n 项和。 $n \in [5, 50]$

样例输入：2

样例输出：3.50000

P1027.Problem (while do...while)

用二分法求 $2x^3-4x^2+3x-6=0$ 在 $(-10,10)$ 的根。(二分法高中有的，不会的童鞋自己去网上找吧)最后的精度到达0.01即可

P1028.Problem (for&&for) page 86 title19

输出所有的水仙花数。水仙花数是它本身恰好等于他各个位数上的立方和的三位数。如 $153=1^3+5^3+3^3$ ，153是水仙花数。

P1029.Problem (*非递归斐波拉契数列)

P1030.Problem (*非递归辗转相除)

Level 4 数组

字符数组、字符串没有题目。

P1031.Example (一维数组)

输出斐波拉契数列前n项：0,1,1,2,3,5,8,13,21,34,55

样例输入：6

样例输出：0,1,1,2,3,5

P1032.Example (多维数组)

输入一个 $n*m$ 的矩阵，求出各列的和。

样例输入：

3 5

1 2 3 4 5

2 3 4 5 6

3 4 5 6 7

样例输出：

6 9 12 15 18

P1033.Problem (array[])

输出三级斐波拉契数列前n项：0,1,1,2,4,7,13,24,44

样例输入：6

样例输出：0,1,1,2,4,7

P1034.Problem (array[]) 陶陶摘苹果，题目百度去

陶陶家的院子里有一棵苹果树，每到秋天树上就会结出10个苹果。苹果成熟的时候，陶陶就会跑去摘苹果。陶陶有个30厘米高的板凳，当她不能直接用手摘到苹果的时候，就会踩到板凳上再试试。现在已知10个苹果到地面的高度，以及陶陶把手伸直的时候能够达到的最大高度，请帮陶陶算一下她能够摘到的苹果的数目。假设她碰到苹果，苹果就会掉下来。

样例输入：

100 200 150 140 129 134 167 198 200 111

110

样例输出：5

P1035.Problem (array[][])

输入一个n*m的矩阵，求出各斜行的和。

样例输入：

5 3

1 2 3 4 5

2 3 4 5 6

3 4 5 6 7



样例输出：

3 6 9 12 15 10 5

P1036.Problem (array[])

输入n，输入n个数，输出其中最大的数以及最大数的序号

样例输入：

8

5 180 8 7 9 4 13 2

样例输出：

180 2

P1037.Problem (*选择排序)

P1038.Problem (array[])

输入n，输入n个排好序的数，在输入一个带插入的数，将这个数插入此序列中，并保持从小到大的顺序输出

样例输入：

7

2 5 7 8 9 13 180

4

样例输出：

2 4 5 7 8 9 13 180

P1039.Problem (*插入排序)

P1040.Problem (array[])

输入n,m，分别输入n个，m个排好序的数，将这2个数列合并，并保持从小到大的顺序输出

样例输入：

7 5

2 5 7 8 9 13 180

3 4 10 11 12

样例输出：

2 3 4 5 7 8 9 10 11 12 13 180

P1041.Problem (*归并排序)

P1042.Problem (array[])

输入n，输入n个数，存入数组后，将数组内的数交换为倒序存储。

样例输入：

8

2 4 5 7 8 9 13 180

样例输出：

180 13 9 8 7 5 4 2

P1043.Problem (*冒泡排序)

P1044.Problem (*二分查找)

P1045.Problem (array[][])

输入杨辉三角的前n行。

样例输入：8

样例输出：

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

1 5 10 10 5 1

1 6 15 20 15 6 1

1 7 21 35 35 21 7 1

P1046.Problem (*数列)

Level 5.1 函数

函数返回值、参数类型、引用没有题目

P1047.Example (function) 写一个求一个数阶乘的函数, 在主程序中调用。输入n, 求 $1!+2!+\dots+n!$

样例输入：3

样例输入：9

P1048.Example (function的多级调用) $\sinh(x)=(e^x-e^{-x})/2$ 求 $\sinh(e^{\sinh(x)})$ 可以使用cmath库中的exp()函数

P1049.Example (递归) 使用递归求n!

P1050.Problem (function) 写一个求素数的函数。在主程序中调用这个函数，求出结果。

样例输入1: 97

样例输出1: Yes,it is.

样例输入2: 99

样例输出2: No,it isn't.

P1051.Problem (function) 写一个将字符串转换为数字的函数。在主程序中调用这个函数，求出结果。注意，在主程序中用字符串String读入，用这个函数转换成数字后，输出。（）

样例输入：94375732597329573297592

样例输出：94375732597329573297592

P1065.Problem @高精度 (*高精度加法)

P1066.Problem @高精度 (*高精度减法)

P1067.Problem @高精度 (**高精度乘法)

P1052.Problem (function) 写一个判断一个年份是不是闰年的函数leap()。在主程序中调用这个函数，求出结果。

样例输入1: 2800

样例输出1: Yes,it is.

样例输入2: 2700

样例输出2: No,it isn't.

P1053.Problem (function&&功能设计) 输入一个日期, 计算出自从2000.1.1到这一天已经过了多久

样例输入: 2000.1.2

样例输出: 2

P1054.Problem (function 递归) 计算如下函数的值。Page127title8 勒让德多项式

$P_n(x)=1, n=0$

$P_n(x)=x, n=1$

$P_n(x)=1((2n-1)xP_{n-1}(x)-(n-1)P_{n-2}(x))/n, n>1$

样例输入: 1 23

样例输出: 23

P1055.Problem (*function 递归) 二分查找。

P1056.Problem (*function 递归) 快速排序

Level *6 简单算法入门

简单程序思想入门及数论基础入门

P1029.Problem @变量轮换 (*非递归斐波拉契数列) 不使用array (数组) 求斐波拉契数列的前n项

样例输入: 5

样例输出: 0,1,1,2,3

P1046.Problem @二进制 @数论 (*数列)

1,3,4,9,10,12,13,27.....

即 $3^0, 3^1, 3^1+3^0, 3^2, 3^2+3^0, 3^2+3^1, 3^2+3^1+3^0, 3^3$

一组以3为底数的幂以及他们的和按照升序排列, 输出这组数列的第n项。

样例输入: 64

样例输出: 729

P1030.Problem @数论 @递推 (*非递归、递归的辗转相除) 辗转相除法求最大公约数。

样例输入: 12 14

样例输出: 2

P1057.Problem @数论 @位运算 (*快速幂) 求 2^n 的值

基本思想:

$$a^{11}=a^{(2^0+2^1+2^3)}=a^{2^0}*a^{2^1}*a^{2^3}$$

$$\text{注意: } a^{2^n}=(a^{2^{n-1}})^2$$

样例输入: 10

样例输出: 1024

P1058.Problem @数论 (*高斯消元) 数学题

P1059.Problem @数论 (*筛法求素数) 输出1...n的所有素数

基本思想：

把从2开始的、某一范围内的正整数从小到大顺序排列，剩下的数中选择最小的数是素数，然后去掉它的倍数。依此类推，直到筛子为空时结束。

样例输入：10

样例输出：2,3,5,7

P1060.Problem @排列组合（*全排列）输入n，输出1...n的所有数的全排列，注意顺序！

样例输入：3

样例输出：

1 2 3

1 3 2

2 1 3

2 3 1

3 1 2

3 2 1

P1061.Problem @排列组合（*下一组排列）输入一组全排列数，输出它的下一组排列（仔细观察上面的顺序）

样例输入：2 1 3

样例输出：2 3 1

简单排序基础入门及查找基础入门

P1044/P1055.Problem @查找法（*二分查找、折半查找）

输入n，输入n个排好序的数，在输入一个待查找的数，查找此数

提示：本题可从顺序查找开始学习

基本思想：

将表中间位置与查找关键字比较，如果两者相等，则查找成功；否则如果中间位置大于查找关键字，则进一步查找前一子表，否则进一步查找后一子表。重复以上过程，直到查找成功，或直到子表不存在为止。

样例输入：

8 4

2 4 5 7 8 9 13 180

样例输出：

yes,the number 4 is in index of 2.

P1037.Problem @排序法（*选择排序）

输入n，输入n个数，将其按照从小到大的顺序排序。（所有排序的题目都是这样的）

基本思想：

选择排序的思想非常直接，要排序么？那好，就从所有序列中先找到最小的，然后放到第一个位置。之后再看剩余元素中最小的，放到第二个位置……以此类推，就可以完成整个的排序工作了。

样例输入：

8

5 180 8 7 9 4 13 2

样例输出：

P1043.Problem @排序法 (*冒泡排序)

依次比较相邻的两个数，将小数放在前面，大数放在后面。如此下去，重复以上过程，直至最终完成排序。

基本思想:

P1061.Problem @排序法 (*桶排序)

当输入是1...n的整数时，可以使用。建立一个n个区间（一个数组），每次读入数据时，放入对应的区间（桶）中，最后，一次扫描区间，有数据则输入，输出的结果显然为有序。

基本思想:

P1041.Problem @排序法 (**归并排序)

1. 以中间为基准将整个数列划分成两个数列。
2. 再对左右区间重复第1步，直到各区间只有一个数，也就是有序（一个数当然有序）。
3. 将上一步划分好的有序的数列合并，保持有序。
4. 重复第三步，直到归并为整个数组。

P1062.Problem @栈 (*括号匹配)

样例输入: (() ()) (Π) ({ }) ()) (() ())

样例输出: No,it isn't.

P1063.Problem @动态规划 (**0-1背包)

基本思想：这个方程非常重要，基本上所有跟背包相关的问题的方程都是由它衍生出来的。所以有必要将它详细解释一下：“将前*i*件物品放入容量为*v*的背包中”这个子问题，若只考虑第*i*件物品的策略（放或不放），那么就可以转化为一个只牵扯前*i-1*件物品的问题。如果不放第*i*件物品，那么问题就转化为“前*i-1*件物品放入容量为*v*的背包中”，价值为*f*[*v*]；如果放第*i*件物品，那么问题就转化为“前*i-1*件物品放入剩下的容量为*v-c*的背包中”，此时能获得的最大价值就是*f*[*v-c*]再加

上通过放入第*i*件物品获得的价值*w*。

样例输入：

3 表示物品的数量

4 3 重量、价值

4 3

5 5

8 背包的最大重量

样例输出：6 可以获得的最大价值，取前两件物品

P1064.Problem @动态规划（**数塔）

给定一个数塔，其存储形式为如下所示的下三角矩阵。在此数塔中，从顶部出发，在每一节点可以选择向下走还是向右走，一直走到底层。请找出一条路径，使路径上的数值（都是正数！这个条件很重要）和最大。

基本思想：**Stept1**：存储信息，将数塔数据存放到二维数组data[][]中。**Stept2**：阶段划分，对于数塔问题应该从上而下逐层决策。首先对第四层的每个数据都进行考虑，选出最优解，然后逐层向上决策，这样逐层递推求出最后结果。**Stept3**：最优解和路径的存储，用maxvalue[][]存储各个路径的最优值，用path[][]存储路径。Maxvalue[i][j]初始化为data[i][j]，Path[i][j]初始化0，Path[i][j]=0为向左，等于1为向右。

样例输入：

3 表示数塔有3层

1

3 2

1 1 5

样例输出：8

简单模拟基础入门

高精度运算基本思想：高精度算法，属于处理大数字的数学计算方法。在一般的科学计算中，会经常算到小数点后几百位或者更多，当然也可能是几千亿几百亿的大数字。一般这类数字我们统称为高精度数，高精度算法是用计算机对于超大数据的一种模拟加,减,乘,除,乘方,阶乘,开方等运算。对于非常庞大的数字无法在计算机中正常存储，于是，将这个数拆开，拆成一位一位的，或者是四位四位的存储到一个数组中，用一个数组去表示一个数字，这样这个数字就被称谓是高精度数。高精度算法就是能处理高精度数各种运算的算法。

P1065.Problem @高精度（*高精度加法）

P1066.Problem @高精度（**高精度乘法）