

写给三年级小孩的 c++ 教程

董映 著

前言

这本书是一位三年级小朋友写给三年级小朋友的 c++教程，作者熟知 c++语法，为帮助同学，编写此书。

本书以三年级小朋友的角度，幽默的语言，以算法竞赛为目标编写。全书以行走中为背景编写。

感谢：

1. [洛谷](#) 提供题库
2. 李想 提供知识教育
3. 房辉&董伟 提供我

如有网络，推荐从 [doblog](#) 官网（吾之博客）阅读。

最后祝大家阅读愉快！

目录：

1. [前言](#)——1
2. [收拾包袱，准备出发](#)——3
3. [不当哑巴](#)——7
4. [数据的小房子](#)——8
5. [也不当聋子](#)——9
6. [学会拐弯](#)——10
7. [反反复复](#)——12
8. [排队](#)——13
9. [小包袱](#)——14

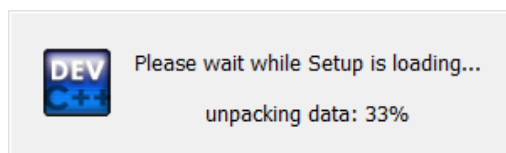
第一章 收拾包袱，准备出发

这是一本地图，我们的目标是在中国靠海边界开始，从水路走到一个神秘岛屿，途中我们会经历许多岛屿，好了，话不多说，出发！

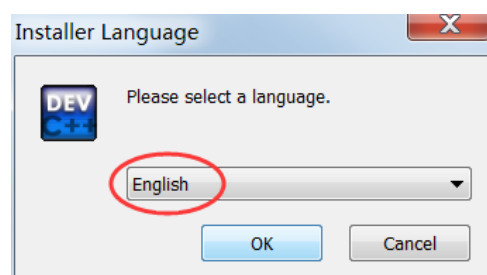
在开始旅途之前，让我们先收拾一下包袱。武器 1——dev-c++!

武器 1——dev-c++!

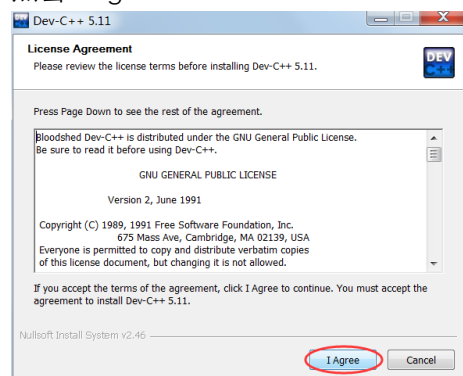
Dev-c++是一个编写 c++代码的东东，众所周知，编程需要代码，但咱打代码需要工具，不然的话，我们对着一个空荡荡的桌面，咱打啥呀？第一步，[下载](#) dev-c++（别告诉我 你连点击都不会），点击立即下载。等待下载完后，把下载的文件打开。首先看到的是加载（要有耐心，没有一分钟）



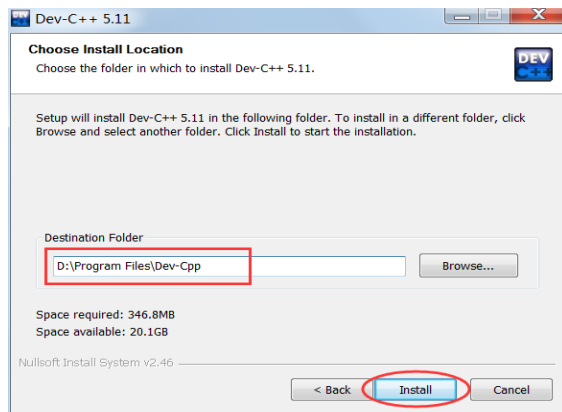
需要选择语言，这里没有中文，不用选，默认英文，点击 OK



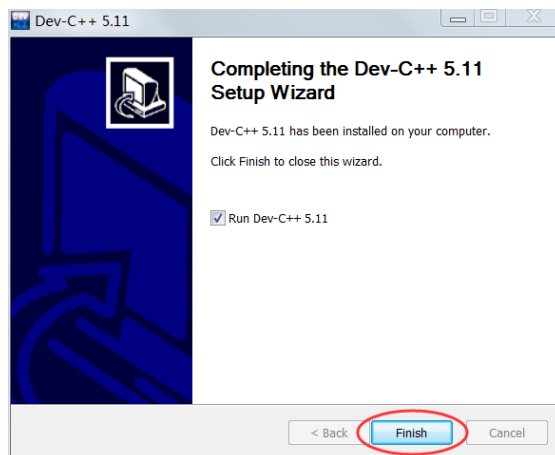
点击 I Agree



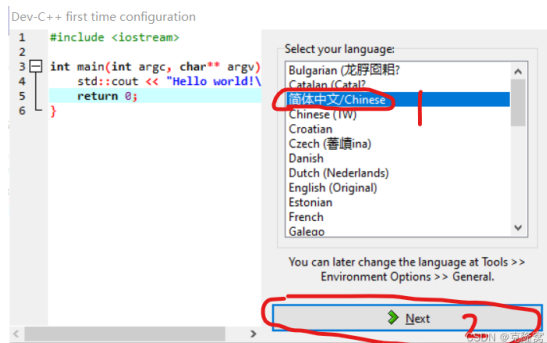
选择 dev-c++ 的居住地，不能带中文。然后点击 Install



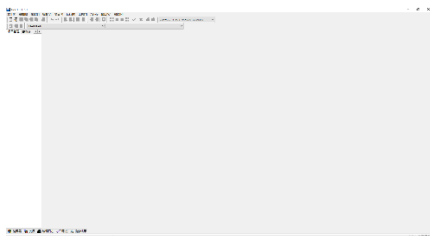
当显示如下画面时，点击 Finish。



此时会弹出下面这个界面，选择简体中文，点击 Next。



再点击 Next,点击 OK，出现下面的界面即可。



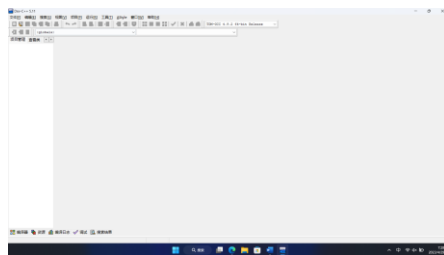
武器 2——洛谷！

打开 luogu.com.cn, 点击右上角注册，一定要填完所有信息（包括验证码信息），勾选

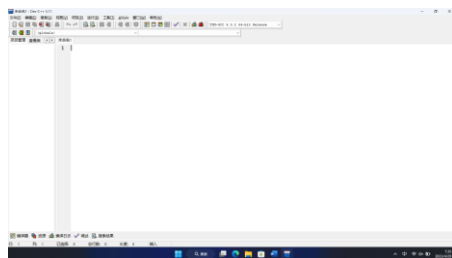
同意《洛谷用户协议》，完成注册。

每当我们学完一个新知识时，我会给出一些练习题，这是，请按照下面的步骤完成练习题：

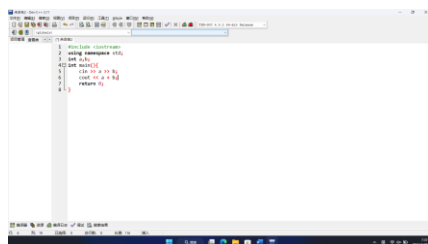
1. 首先，打开 dev-c++。



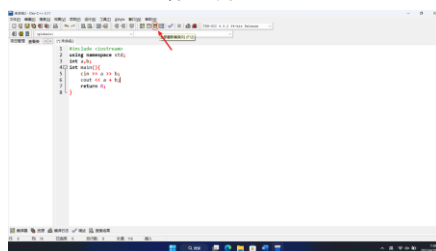
2. 然后，按住 ctrl 键，别松手，另一只手按下 n，新建一个文件。



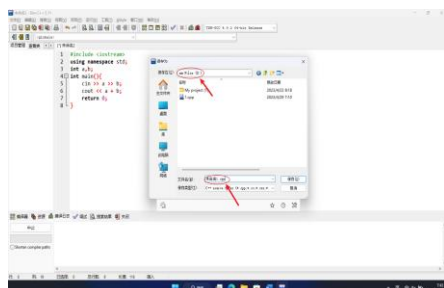
3. 接着，在其中输入代码（下图是一个简单的例子）。



4. 按下编译并运行按钮。



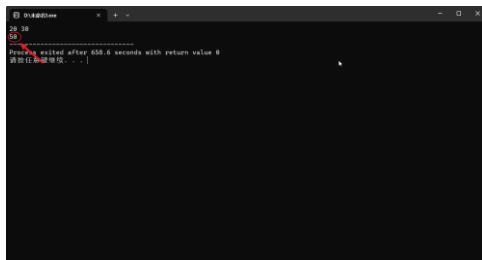
5. 他会让你把代码保存，选择一个你喜欢的位置，设置文件名，点击保存。



6. 如果下方没有红色的字，说明没有错误，此时会弹出一个黑窗，这时打开洛谷，找到问题跳转下面的输入框，输入我给你的题号，点击跳转，这里拿 P1001 举例。



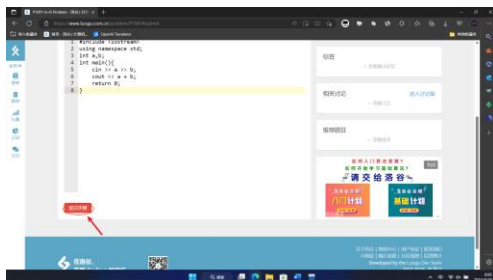
7. 向下滑动，找到输入 #1，点击复制，黏贴到刚才的黑色窗口中，按下回车 (enter) 如果它显示的结果和输出 #1 相符，那么你的程序有可能对了，比如下面这个可能对了。这时，按下空格键（就是那个大长条）退出。



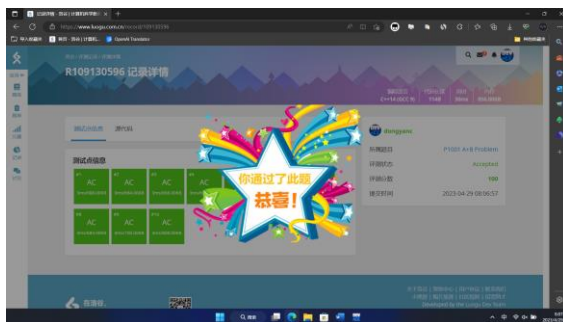
8. 回到 dev-c++，把代码复制一下，来到洛谷，向上翻，点击提交答案。



9. 将代码黏贴到代码框中，然后向下滑，点击提交评测。



10. 等待一会儿，如果显示“你通过了此题，恭喜”，那么恭喜你，通过了此题。



否则的话，仔细检查你的代码，一定有你没想到的地方。

好了，让我们从零开始，快乐的学习 c++ 吧！

第二章 不当哑巴

从这一章开始，我们就要开始冒险了。准备好了吗？Ready？Go！

来到第一个岛屿，叫做说话岛，这是什么，我们来一探究竟。

对于我们人，说谁哑巴谁都不愿意，计算机也是这样，让人家当哑巴人家也不愿意呀！所以，这个岛屿给我们的任务就是让计算机能说。

其实计算机天生是会说的，只是得把这个功能启动，就用下面两行代码。

```
1. #include <iostream>
2. using namespace std;
```

其中“iostream”是值得输入输出功能，#include<xxx>表示将 xxx 功能启动，using namespace std 这个是什么意思你不用管，只知道需要输入他就行了。

这时，我们需要使用这个功能，否则的话，就和你买了一个好玩的玩具，但你不玩，有什么用？首先得放入一个代码隧道，只有放入隧道里的代码才会被执行，这个隧道叫做 main。

```
1. int main(){
2.     return 0;
3. }
```

我们的代码就要放在“{”和“return 0;”之间。最重要的一步来了，这时候，我们要让他说话啦！比如，我们让他说个“hello”吧！

```
1. cout << "hello";
```

完整代码：

```
1. #include <iostream>
2. using namespace std;
3. int main(){
4.     cout << "hello";
5.     return 0;
6. }
```

这样，我们的计算机就会说话啦！

备注：c++的每一个语句后面需要加一个;就像我们的句号一样，一定要记住！

练习：B2002 U291288

第三章 数据的小房子

我们知道，程序的本质就是 0 和 1，把 0 和 1 串起来就是数据，那么数据存在哪里呢？他们就和我们人一样，要住在房子里。那么他们的房子是什么呢，就是变量。

变量有许多种类型，比如，字符，数字。。。。。下面，我们来认识一下它们。

首先登场的是数字！数字分整型（即整数）和浮点型（即小数）

整型：

有 int, long long, 和 unsigned long long，他们就像不同大小的房子，能住不同数量的人，int 可以存一个 -2147483648~2147483647 的整数，比如 1, 2, 5, 100, 10000。long long 可以存一个 -9223372036854775808~9223372036854775807 的整数。unsigned long long 可以存一个 0~18446744073709551615 的整数。

浮点型：

一般 double 就可以了，小数点后 15 位。

字符：

char, 可以储存一个英文字母，或符号（也就是键盘上能看见的这些）。

定义变量的语法是

```
1. 变量类型 变量名；
```

或者直接带上初始值

```
1. 变量类型 变量名 = 值；
```

后续可以在对它进行修改

```
1. 变量名 = 值；
```

数字型变量或数值可以进行运算

+: 加法 -: 减法 *: 乘法 /: 除法 %: 取模 (a%b 即为 a/b 的余数，只能对整型进行运算) 他们的运算顺序是：有括号先括号，没括号先算*/%，再算+-。

如果这里有一个 int, 想把它变成 long long 怎么办？假设这个变量叫 x，那么可以这样写

```
1. (long long)x;
```

或者这样：

```
1. long long(x);
```

这里如果我要更改这个变量的话，可以这样：

+:

```
+1: ++x;或 x++;
```

```
+n: x += n;
```

-: 类似于加

```
*: *n: x *= n;
```

/: 类似于乘

好了，又到了练习时间了。这回，我亲自来考你！

题目要求计算 200 和一亿的积。试一试吧，把答案与计算器中的答案比较一下找出你的错，我相信你一遍绝对对不了，这个得用 long long，代码：

```
1. #include <iostream>
2. using namespace std;
```

```

3. int main(){
4.     cout << (long long)200 * (long long)100000000;
5.     return 0;
6. }

```

知道自己错在哪里了吧，没关系，犯错不怕，改了就好，没有一个人没犯过错。

第四章 也不当聋子

上一次我们让计算机不当哑巴，这回我们让他不当聋子。

我们知道，变量是储存数值的，那么，我们就让听到的东西存进变量里。

输入的命令长这样：

```
1. cin >> 变量1 >> 变量2 >> 变量3 >> ... >> 变量N;
```

其中，如果变量 x 是 `char` 类型，那么他只能输入字符，其他类似，所有数据之间在输入时以空格隔开（`char` 除外），比如运行下面代码时，如果输入为“1 2 3”，那么， $x=1, y=2, z=3$

```

1. int x,y,z;
2. cin >> x >> y >> z;

```

好了，又到了练习时间，今天的题目是：B2001, B2003, B2005, B2008, B2009, B2010, B2020, B2028 加油吧，少年！保送不是梦，但不容易。这里给出 B2028 的代码，仅供参考

```

1. #include<iostream>
2. using namespace std;
3. int main(){
4.     char a,b,c;
5.     cin >> a >> b >> c;
6.     cout << c << b << a;
7.     return 0;
8. }

```

第五章 学会拐弯

上一章我们学习了输入，那么，这一次我们来学习根据输入的数据来进行判断！判断长这样：

```
1. if (条件){  
2.     代码  
3. }
```

常见的条件如下：

写法	成立条件
True	永远成立
False	永远不成立
N	$n \neq 0$
$a \&\& b$	$a = true \text{ and } b = true$
$a b$	$a = true \text{ or } b = true$
!a	$a = false$
$a == b$	$a = b$
$a != b$	$a \neq b$
$a <= b$	$a \leq b$
$a >= b$	$a \geq b$

代码内容里可以填任何内容。

比如，判断一个数是否是偶数可以这样：偶数的特点就是能被 2 整除，也就是 $x \% 2 == 0$ ，这样，我们可以写出代码(“even”是 even number 偶数的第一个单词)：

```
1. int x;  
2. cin >> x;  
3. if (x % 2 == 0){  
4.     cout << "even";  
5. }
```

另外，我们还有两种判断语句：

1. if...else...

格式：

```
1. if (条件){  
2.     代码 1  
3. }  
4. else{  
5.     代码 2  
6. }
```

else 的意思是“除此之外”，执行流程：如果 if 条件成立，那么执行代码 1，否则，执行代码 2。

2. if...else if...else:

格式：

```
1. if (条件 1){  
2.     代码 1
```

```
3.  }
4.  else if (条件 2) {
5.      代码 2
6.  }
7.  else if (条件 3){
8.      代码 3
9.  }
10. ...
11. else{
12.     代码 n
13. }
```

else if 可以有无限个，else 可有可无，执行顺序是：如果满足条件 1，那么执行代码 1，并退出整个流程，否则，如果满足条件 2，那么执行代码 2，并退出整个流程…以此类推。如果全都不满足，如果有 else 部分，那么执行 else 对应的代码。

补充一句，所有判断均可嵌套，比如 if 里面还是一个 if。

好了，这就是学会拐弯的内容了，下面我们来练习：

B2035, B2036, B2037, B2039, B2040, **B2043**。

第六章 反反复复

机器人最擅长干的事儿就是重复了，那么重复在 c++ 中是什么呢？我们来学习一下循环 (loop)。

循环有两种，我们先来学习 for 循环(for loop):

格式：

```
1. for (变量定义;循环条件;变量更新){
2.     代码
3. }
```

在变量定义时定义一个变量（一般为 i,j,k...）需要给定一个初始值，比如 `int i = 1;` 循环条件就是在每循环一次时，如果满足循环条件，那么执行代码，否则退出，比如 `i <= 5`；变量更新指在执行完一遍代码后执行的内容，比如 `++i`；完整示例（输出 100 遍“wa”）：

```
1. for (int i = 1; i <= 100; ++i){
2.     cout << "wa ";
3. }
```

注意！for 可以嵌套，但变量名不可重复！

While 循环：

格式：

```
1. while (条件){
2.     代码
3. }
```

流程很简单，只要满足条件就执行一遍代码，比如将上面的例子转化为 while 循环：

```
1. int x = 1;
2. while (x <= 100){
3.     cout << "wa ";
4.     ++x;
5. }
```

While 循环也可嵌套，for 和 while 也可互相嵌套

循环还有两个知识：continue&break!

continue:当我循环到某一遍时，需要这一遍结束，继续后面的内容时，我们就用一句 continue 来解决，大概明白了吗？

Break:直接结束整个循环。

洛谷：哈哈，又要来找我啦！今天，我来给出题目！题目：B2057，B2058，B2059，B2061。

第七章 排队

上一章我们学习了循环，这时，岛上的一个公民问了一个问题：我这里有一串数，你能倒着在给我吗？比如：1 5 6 2 4 7 3 10，结果就是 10 3 7 4 2 6 5 1，像这样需要存一串类似的东西，我们就用——数组！话不多说，上格式！

1. 变量类型 数组名[长度];

意思就是创建一个这个长度个这个类型的变量，假设数组名叫 a，长度为 10，那么这 10 个变量分别是 a[0],a[1],a[2],a[3]...a[9]，**注意，从 0 开始，到长度-1!!!** 长度有个学问，一般比需求要大 100，一定记住。这样，结合循环，我们可以解决刚才这个问题：给定第一行一个 n,第二行 n 个数，要求倒序输出这些数。

```
1. cin >> n;
2. for (int i = 1; i < n; ++i){
3.     cin >> a[i];
4. }
5. for (int i = n; i >= 1; --i){
6.     cout << a[i];
7. }
```

这一章太短了，但是没办法，数组就这么简单，好了，某谷出题时间：B2089，B2098（这道题是不是很难？哈哈，答案就像海绵里的水，挤挤总是有的）。

第八章 小包袱

上一章我们学习了可以存多个类似的东西的数组，那么这回，公民有来找我们麻烦了，这回，麻烦是：这里有 n 个人，每个人有身高(double)，体重 (double)，年龄 (int) 这该怎么办？这里我们来学着定义一种类型。被称为——结构体：

```
1. struct 类型名{  
2.     定义变量;  
3. }
```

定义变量时我们可以定义各种类型的各种变量。比如上面这个可以：double weight(体重), height(身高);int age(年龄)，然后我们可以拿着这个类型去定义数组，变量。比如：

```
1. struct person{  
2.     double weight,height;int age;  
3. }  
4. person p[100];
```

这里，如果我们想获得第十个人的年龄，就用 `p[10].age`，其他类似。也可以当成变量输入。这回，练习题我亲自出：给出人数 n ,每人有身高，体重，年龄。请求出年龄最小的人，并输出他的身高，体重，年龄。

好了，到这里，本书就结束了，最后，留一个有奖竞猜活动。本书里某个代码有 bug，找到者，发送电子邮件至 admin@dododo.eu.org，请给出：页面，错误点，正确者，赠送大公司正版标准 c++教程阅读 1 个月，大家加油吧！