



格式化输入输出和流输入输出

在 C++语言中，cin / cout 和 scanf / printf 都是输入/输出功能。

我们对其进行比较：

	cin / cout	scanf / printf
来源	C++输入输出流	C 语言标准输入/输出库函数
引用库	<iostream>	<cstdio>
endl	支持	不支持
默认效率	低	高
数据格式化	弱	强
变量数据类型自动识别	能	不能

表1. 输入/输出功能对比



例：cout 和 printf

对比下面两段代码，观察有什么不同。

【参考程序 1】

```
1. #include<iostream>
2. using namespace std;
3. int main()
4. {
5.     cout << "Hello World!" << endl;
6.     return 0;
7. }
```

【参考程序 2】

```
1. #include<stdio>
2. int main()
3. {
4.     printf("Hello, World!\n");
5.     return 0;
6. }
```

【分析】

第一行引入的库是不同的。这是因为两个程序采用了不同的输出方式：

- cout 是 C++语言才有的输出方式，需要引入<iostream>;
- printf 是 C 语言就有的输出方式，需要引入<stdio>。

只使用 C 语言的功能，例如 printf，可以不写 using namespace std。

endl 和\n 的意义一致，但是 endl 要引入<iostream>。

平时编程，我们使用哪一种都是可以的。



例：cin 和 scanf

【问题描述】

输入若干对整数，输出每对之和。

假设每个整数不超过 10^9 ，一共不超过 10^6 个数对。

【参考程序 1】

```
1. #include<stdio>
2. int main()
3. {
4.     int a, b;
5.     while(scanf("%d%d", &a, &b)==2)
6.     {
7.         printf("%d\n", a + b);
8.     }
9.     return 0;
10. }
```

【参考程序 2】

```
1. #include<iostream>
2. using namespace std;
3. int main()
4. {
5.     int a, b;
6.     while(cin >> a >> b)
7.     {
8.         cout << a + b << endl;
9.     }
10.    return 0;
11. }
```