

1-3 “Hello World——输入输出进阶

C++与C语言的输出方式

1. C语言中的输入输出函数是scanf和printf，C语言风格的输入输出方式的核心是使用%开头的占位符来指定输入输出格式（如保留多少位浮点数、输出的数据占多宽的空间）。
2. C++的输出方式稍显复杂，**同时在批量处理大数据存在性能问题**，所以视情况而定选择不同的输入输出方式

常见输入输出占位符

一个占位符格式一般为——“%[符号][宽度][.精度]类型”

1. [宽度]：表示最少输出的字符个数
2. [符号]：“-”表示对齐方式
 - a. %-8,左对齐，当显示字符不足8时，右补空格
 - b. %08,右对齐，当显示字符不足8时，左补0
3. [.精度]对于浮点数表示小数点后的位数
4. 数值小数点后的位数大于显示精度，则只能显示[精度]个有效位数（四舍五入），如果数值小数点后的位数小于显示精度，则补零。
5. **类型**
 - a. **%d整数int、%lld整数long long**,
 - b. **%o整数的8进制形式、%x%X整数的16进制形式**
 - c. **%f浮点数float【输入】、浮点数float/double【输出】，%lf浮点数double【输入】**
 - d. **%c字符char、%s字符串**

C语言风格的输入输出函数

1. 输入函数scanf
 - a. 一个变量：scanf("%d",&num); scanf("%f",&f);
 - b. 多个变量：scanf("%d%d%d",&a,&b,&c); scanf("%d%s",&age,name);【注意字符数组不用&】
2. 输出函数printf
 - a. 字符串：printf("Hello world!\n"); printf("str = %s\n",str); // 输出字符串

- b. 数字: `printf("The sum of 2 and 3 is %d\n", 2+3);` `printf("The value of pi is %.2f\n", 3.14159);`
- c. 字符串: `printf("My name is %s and I am %d years old.\n", "John", 25);`

上手练练

代码块

```
1  #include <bits/stdc++.h>
2  using namespace std;
3  int main() {
4      // 声明变量
5      int num;
6      float f;
7      int a, b, c;
8      int age;
9      char name[50]; // 字符数组, 用于存储字符串
10     double pi = 3.14159;
11
12     // 使用scanf函数读取输入
13     printf("plz input the num: ");
14     scanf("%d", &num); // 读取一个整数并存储在num中
15
16     printf("plz input the float: ");
17     scanf("%f", &f); // 读取一个浮点数并存储在f中
18
19     printf("plz input a b c: ");
20     scanf("%d%d%d", &a, &b, &c); // 依次读取三个整数并分别存储在a、b、c中
21
22     printf("tell me your age and name: ");
23     scanf("%d%s", &age, name); // 读取一个整数和一个字符串, 存储在age和name中
24     // 注意: 对于字符数组, 不需要使用&符号
25
26     // 使用printf函数信息
27     printf("Your num is : %d\n", num); // 整数num
28     printf("The float you entered is: %.2f\n", f); // 浮点数f, 保留两位小数
29     printf("The three integers you entered are: %d, %d, %d\n", a, b, c); // 三个
    整数a、b、c
30     printf("Your age is: %d, name is: %s\n", age, name); // 年龄和名字
31
32     // 带有格式控制的数字
33     printf("The value of pi is: %.5f\n", pi); // 保留五位小数
34     printf("The value of pi is: %08.5f\n", pi); // 保留五位小数, 总宽度为8, 不足用0
    填充
35     printf("The value of pi is: %-8.5f\n", pi); // 保留五位小数, 总宽度为8, 不足用空
    格填充, 左对齐
```

```
36
37 // 整数的八进制和十六进制形式
38 printf("The octal form of num is: %o\n", num); // num的八进制形式
39 printf("The hexadecimal form of num is: %x\n", num); // num的小写十六进制形式
40 printf("The hexadecimal form of num is: %X\n", num); // num的大写十六进制形式
41 return 0;
42 }
```