C语言中的 `printf` 函数通过**占位符**（格式说明符）将数据按指定格式输出。以下是常见占位符的使用方法和示例：

基础占位符

占位符	用途	示例	输出结果
`%d`	有符号十进制整数	`printf("%d", 123);`	`123`
`%u`	无符号十进制整数	`printf("%u", 255);`	`255`
`%f`	浮点数（默认6位小数）	`printf("%f", 3.1415);`	`3.141500`
`%c`	单个字符	`printf("%c", 'A');`	`A`
`%s`	字符串	`printf("%s", "Hello");`	`Hello`
`%x`	十六进制小写	`printf("%x", 255);`	`ff`
`%X`	十六进制大写	`printf("%X", 255);`	`FF`
`%o`	八进制整数	`printf("%o", 64);`	`100`
`%p`	指针地址	`printf("%p", &a);`	`0x7ffeeb5b9a2c`
`%%`	输出百分号	`printf("%%");`	`%`

格式修饰符

1. **精度控制**：
   - 浮点数保留小数位数：`%.2f`。
   - 字符串截断：`%.5s`。

```c
printf("%.2f", 3.1415926);  // 输出 3.14
printf("%.5s", "Hello World"); // 输出 Hello
```

2. **宽度和对齐**：
   - 最小宽度：`%5d`（右对齐，补空格）。
   - 左对齐：`%-5d`。
   - 前导填充0：`%05d`。

```c
printf("%5d", 10);      // 输出 "   10"
printf("%-5d", 10);     // 输出 "10   "
printf("%05d", 10);     // 输出 "00010"
```

3. **科学计数法**：

- `%e`（小写）或 `%E`（大写）。
- 强制指数符号：`%+.2e`。

```
printf("%e", 1234.56);    // 输出 1.234560e+03
printf("%.2E", 1234.56); // 输出 1.23E+03
```

4. **其他修饰符**：

- `%#x`：输出十六进制前缀 `0x`。
- `%+d`：显示正负号。

```
printf("%#x", 255);      // 输出 0xff
printf("%+d", 100);      // 输出 +100
```

长度修饰符

修饰符	含义	示例
`h`	短整型（`short`）	`%hd` → `short`
`l`	长整型（`long`）	`%ld` → `long`
`ll`	长长整型（`long long`）	`%lld` → `long long`
`L`	长双精度（`long double`）	`%Lf` → `long double`

```
long num = 123456L;
printf("%ld", num);         // 输出 123456

long double pi = 3.1415926535L;
printf("%.2Lf", pi);        // 输出 3.14
```

错误示例和注意事项

1. **类型不匹配**：

```
float f = 3.14;
printf("%d", f);  // 错误：类型不匹配，输出无意义值
```

2. **悬空参数**：

```
printf("%d %d", 10);  // 参数不足，导致未定义行为
```

3. **无效指针：**

```
int x = 65;
printf("%s", x);  // 错误：x被当作字符串地址，可能崩溃
```

完整示例

```
#include <stdio.h>

int main() {
    int a = 123;
    float b = 45.6789;
    char c = 'Z';
    char str[] = "Hello World";
    long double d = 3.1415926535L;

    printf("十进制整数：%d\n", a);
    printf("浮点数（两位小数）：%.2f\n", b);
    printf("科学计数法：%.3e\n", b);
    printf("字符：%c\n", c);
    printf("字符串（截断前5字符）：%.5s\n", str);
    printf("十六进制整数：%#x\n", a);
    printf("长双精度：%.2Lf\n", d);

    return 0;
}
```

输出：

```
十进制整数：123
浮点数（两位小数）：45.68
科学计数法：4.568e+01
字符：Z
字符串（截断前5字符）：Hello
十六进制整数：0x7b
长双精度：3.14
```

通过正确使用占位符和修饰符，可以灵活控制输出格式，确保程序的输出符合预期。