

编程语言

函数式编程

编译器

类型系统

编程语言比较

✎ 修改

关注者

200

被浏览

20,059

## 如何简单的实现类型安全的 printf?

✎ 修改

完整的问题描述参见：[pdfs.semanticscholar.org/...](https://pdfs.semanticscholar.org/...)

✎ 修改

他们也关注了该问题



关注问题

✎ 写回答

+ 邀请回答

💬 添加评论

🚩 分享

★ 邀请回答

🚩 举报

...

查看全部 15 个回答



**Felis sapiens** ★  
函数式编程、编程语言、编程 话题的优秀回答者

12 人赞同了该回答

两类思路，要么用依赖类型来造，格式串在类型层面有表示，然后用这个类型计算出剩下的函数类型和实现；另一类就是元编程了，对每个不同的格式串在当前调用处特化一个 printf。

发布于 2018-08-04

▲ 赞同 12 ▼

💬 6 条评论


🚩 分享

★ 收藏

❤ 感谢

...

更多回答



**dram**  
dram ♥ Bad Apple!!

55 人赞同了该回答

**2018-08-08 update:** 好像有人特别想要 delimited continuation，那就如你所愿；往后翻吧。

在这个问题的回答底下评论里我说 Idris 可以做一个**格式字符串可以运行时确定的**，而且是**编译时静态类型安全**的 printf，感觉是时候兑现一下诺言了：

更新：好像评论区有人不太明白，以下这些**全部都是静态检查的类型**，如果哪儿写的不对是**编译时检查出来，不是运行时检查**。有不觉得有区别的看一下最底下的那个 main 里面什么叫运行时。

```
module Main

%default total
```

这里我们参照 [github.com/mukeshtiwari...](https://github.com/mukeshtiwari...) 实现一个 printf，但是为了方便使用有所修改。

printf 的格式串由 %d 这样的参数段和正常字符的字面量段组成，我们定义一些相关的类型

展开阅读全文 ▼

▲ 赞同 55 ▼

💬 16 条评论

🚩 分享

★ 收藏

❤ 感谢

...



**6hu2t32**  
今天我们都是凡学家

68 人赞同了该回答

C++ 17只要用自定义literal，把字符串literal搞到类型里去就可以了，这有什么难的？



关于作者



**Felis sapiens**

★ 函数式编程、编程语言、编程 话题的优秀回答者

👤 电影旅行敲代码、Antokha Yuuki、暮无井见铃也关注了她

回答

624

文章

40

关注者

14,871

已关注

发私信

被收藏 1 次

Software

道穷则变 创建

0 人关注

相关问题

为什么尾递归优化似乎不常出现在非函数式语言的解释器、编译器中？ 9 个回答

Rust的设计中为什么要区分不可变变量和常量？ 12 个回答

弱类型、强类型、动态类型、静态类型语言的区别是什么？ 31 个回答



```
#include <cstdio>
```

```
template<typename...> struct LIST {};
```

```
template<typename, typename> struct CONS;
```

```
template<typename T, typename... U>
```

```
struct CONS<T, LIST<U...>> {  
    using TYPE = LIST<T, U...>;  
};
```

```
template<typename T, typename U>
```

[展开阅读全文](#)[赞同 68](#)[14 条评论](#)[分享](#)[收藏](#)[感谢](#)[查看全部 15 个回答](#)[C语言编译器为什么能够用C语言编写?](#)[个回答](#)[哪些学校的类型系统与编译器方向不错?](#)[8 个回答](#)

## 相关推荐



**淼懂物理学：理解世界的极简指南**

共 31 节课

[试听](#)

**数学妙啊！妙！**

张英锋 等

289,059 人读过

[阅读](#)

云数据库MySQL基础版**1元**体验

为 中小企业 量身打造，单节点架构，同等配置性能超业界 300%

[立即抢购](#)

腾讯云MySQL数据库**1元**体验



支持主从实时热备,自动容灾,备份,恢复,监控,迁移等数据全套功能。满足入门学习,小规模应用场景



[刘看山](#) · [知乎指南](#) · [知乎协议](#) · [隐私政策](#)

[应用](#) · [工作](#) · [申请开通知乎机构号](#)

[侵权举报](#) · [网上有害信息举报专区](#)

[违法和不良信息举报](#): 010-82716601

[儿童色情信息举报专区](#)

[电信与服务业务经营许可证](#)

[网络文化经营许可证](#)

[联系我们](#) © 2018 知乎

