

python 数据处理编程

卓能文

2023-11-26

目 录

1 简介	3
1.1 概述	3
2 学习	4
2.1 深入	5

图形列表

图 1.1: 苹果	4
图 2.1: 苹果	6

代码列表

代码 1.1: 计算斐波纳契序列	3
代码 1.2: hello.rs	4
代码 1.3: 日志过滤	4
代码 2.1: 计算斐波纳契序列	5
代码 2.2: hello.rs	6

1 简介

希克森写道，Flutter 是“老谷歌人”最后推出的项目之一，属于 Larry Page 在 Alphabet 创建前不久发起的一系列雄心勃勃的实验的一部分。Flutter 团队基本上也像一家初创公司一样运作，而且在很大程度上是建立在年轻的谷歌文化基础上。例如优先考虑内部透明度、工作/生活平衡和数据驱动决策。所以整个团队从一开始就非常开放，这使得他们能够积极构建一个健康的开源项目。

与此同时，谷歌的高速发展让其公司文化受到了侵蚀。

希克森表示：“透明度消失了。以前我会迫不及待地参加每一次公司全体会议，以便了解情况。现在，我发现自己甚至能够预测高管们所给出的每一个答案。今天，我不知道在谷歌还有谁能解释它的愿景，员工士气也空前低落。如果你和湾区的治疗师交谈，他们会告诉你，他们所有的谷歌客户都对谷歌感到不满意。”

对于今年 1 月份的“大裁员”，希克森称这是公司非常短视的决策——目的只是为了确保股价保持季度增长，而不是遵循谷歌以前的战略，即优先考虑长期成功，即使这会导致短期损失。

而且自那以来，谷歌一直在进行规模更小、更低调的裁员，以至于员工们现在都在一份内部文件中追踪裁员情况。

希克森说，这些裁员对公司文化产生了“潜在”影响，可能导致员工“大幅减少冒险行为”。他写道：“我现在在谷歌看到了很多负面现象，比如员工对管理层缺乏信任，而管理层也不再信任员工，公司制定的政策也越来越显得空洞。”

此前曾有媒体报道称，谷歌的普通员工和管理层之间关系紧张。比如，谷歌要求高管为员工表现进行绩效打分。

希克森建议，应该努力将权力从首席财务官办公室转移到这样一个人手中——他应该对如何利用谷歌广泛的资源为用户创造价值有清晰长期的愿景。

至于谷歌的文化是否可以挽救，希克森认为可以，但他称“时间紧迫”。他写道：“谷歌文化的腐化最终将变得不可逆转，因为它需要充当道德指南针的那种人，而这类人在没有道德指南针的情况下不会愿意加入谷歌。”

1.1 概述

代码 1.1：计算斐波纳契序列

```
1 def fibonacci(n):python
2     if n <= 1:
3         return n
4     else:
5         return fibonacci(n - 1) + fibonacci(n - 2)
6
```



图 1.1: 苹果

代码 1.2: hello.rs

```
1 fn main() {  
2     println!("Hello, world!");  
3 }
```

rust

代码 1.3: 日志过滤

```
1 // Print file lines that start with "DEBUG:"  
2 import os  
3  
4 // `read_file` returns an optional (`?string`), it must be checked  
5 text := os.read_file('app.log') or {  
6     // `err` is a special variable that contains the error  
7     // in `or {}` blocks  
8     eprintln('failed to read the file: ${err}')  
9     return  
10 }  
11  
12 lines := text.split_into_lines()  
13 for line in lines {  
14     if line.starts_with('DEBUG:') {  
15         println(line)  
16     }  
17 }  
18
```

2 学习

Fortran 2023 是 Fortran 编程语言的最新版本。它是一种高性能并行编程语言，专为科学和工程计算而设计。Fortran 2023 具有许多强大的特性，使得它成为处理大规模数值数组的快速计算的理想工具。

1. 高性能：Fortran 2023 针对科学和工程计算应用进行了优化，具有成熟且经过实战验证的编译器和库，可以编写接近底层的高效二进制代码。
2. 静态和强类型：Fortran 是一种静态和强类型的语言，这使得编译器能够在早期捕捉到许多编程错误。这也使得编译器能够生成高效的二进制代码。
3. 易学易用：Fortran 是一种相对简单的语言，学习和使用起来非常容易。通过将大多数数学和算术操作表达为方程式，可以轻松地处理大型数组。
4. 多功能：Fortran 允许您以最适合问题的方式编写代码：命令式、过程式、面向数组、面向对象或函数式。
5. 本地并行：Fortran 是一种本地并行编程语言，具有直观的类似数组的语法，用于在 CPU 之间传递数据。您可以在单个 CPU 上、共享内存多核系统上或分布式内存的 HPC 或基于云的系统上运行几乎相同的代码。Coarrays、teams、events 和集体子程序允许您表达最适合当前问题的不同并行编程模式。

2.1 深入

代码 2.1：计算斐波纳契序列

```
1 def fibonaci(n):  
2     if n <= 1:  
3         return n  
4     else:  
5         return fibonaci(n - 1) + fibonaci(n - 2)  
6
```

python



图 2.1: 苹果

代码 2.2: hello.rs

```
1 fn main() {  
2     println!("Hello, world!");  
3 }
```

rust