

让内存不再泄漏的实用技巧[2] PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/234/2021\\_2022\\_\\_E8\\_AE\\_A9\\_E5\\_86\\_85\\_E5\\_AD\\_98\\_E4\\_c103\\_234209.htm](https://www.100test.com/kao_ti2020/234/2021_2022__E8_AE_A9_E5_86_85_E5_AD_98_E4_c103_234209.htm) 性能 该工具的性能非常不错，其GUI设计得也不错(如图1所示)。这个工具直接就可以执行，并且其工作起来无需对源代码进行任何修改。在程序执行时，这个工具会以图形化的方式显示内存的使用情况，以帮助你了解程序运行过程中内存的申请情况（如图1）。图1 Memprof的GUI 该工具目前只能运行于x86和PPC体系结构之上的Linux系统之中。如果你需要用于其它的平台，应该想想使用其它的工具。该工具不是GTK应用程序，所以需要完整的GNOME环境。这样就使得其不能灵活用于所有的地方。此外，该工具的开发工作进展得也比较缓慢(现在是0.4.1版)。结论 如果你喜欢GUI工具并且不介意只能用于Linux以及GNOME之下，该工具应该可以说是非常不错。

工具三：Valgrind Valgrind(<http://developer.kde.org/~sewardj/>)是一个致力于解决所有内存问题的程序，而内存泄漏只不过是其中的问题之一而已。该工具的开发人员是Julian Seward(以Bzip2和Cacheprof而闻名)。该工具宣称自己“是专门致力于解决x86 Linux中开放源代码的内存问题”，事实上，它的确做到了自己的宣言。此外，它还可以描述CPU缓存的使用情况，不过这一功能并不常用。使用的技术 在这个程序中使用的技术非常复杂，不过其文档非常丰富和完整(<http://developer.kde.org/~sewardj/docs/techdocs.html>)。程序分配的每一字节的内存都被一个有九位的状况字跟踪，其目的是用于识别其意图。这种做法大大加重了系统的负担。性

能 这个工具是我们这儿介绍的三款中性能最差的一个，原因是显而易见的。该工具提供的信息细节是三个工具中最丰富的，因而速度也是最慢的。除了一些常见的问题外，该工具还可以发现内存其它的一些问题，甚至一些POSIX线程方面的问题。缓冲的信息对于大部分程序来说似乎没有必要，不过它是一个查看程序性能的很好方式。对于Valgrind来说，值得一提的就是其开发速度非常快，其开发社团也非常活跃。事实上，在Valgrind的主页上作者甚至有一句话：“如果你在使用Valgrind过程中有任何问题，请不要介意，给我发邮件吧”。不过，该工具是专门用于x86的。其界面是纯命令行方式，但是其可用性非常好。该工具可以直接在二进制下运行，所以在使用时并不需要对其进行重新编译。不过要熟练掌握它，还是需要使用者进行一番努力的。此外，虽然该工具曾经使用于Mozilla、OpenOffice等一些大的线程程序，但该工具对线程的支持并不完善。我想如果该工具要是有一个GUI界面，将会赢得更多人的青睐。结论 如果你使用x86，对自己的代码非常了解并且不介意使用命令行方式，那么这个程序将是你的至爱。如果我在此介绍的三款工具你都不喜欢，那也没有关系，可在下面站点中找到很多检测内存错误的工具：[http://www.sslug.dk/emailarkiv/bog/2001\\_08/msg00030.html](http://www.sslug.dk/emailarkiv/bog/2001_08/msg00030.html)。此外，还有一些商业工具，比如Purify、Geodesic等。在此就不详细介绍。100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)