

将Linux代码移植到Windows的简单方法（5）PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/144/2021_2022__E5_B0_86Linux_E4_BB_c103_144609.htm 四：代码运行正常 实际上当link问题解决之后，程序可以在windows环境中运行时，一切就尽在掌握了。如果不考虑做多平台的程序的话，这个时候就可以任意去修改程序了。不过在代码调试过程可能需要一个参照，看看正常的程序运行流程是怎么样的。刚刚移植过来的程序在很多地方并不能马上就能正常的运行。回到Cygwin中，重新编译一个可以调试的版本（在GCC编译选项加上-g3），在需要的时候可以在Cygwin中调试程序。调试可以用GDB或者Insight。如果习惯Windows平台下面编程，可以使用Insight，这是一个TCL/TK脚本程序，它提供了一个Windows界面以方便用户调试程序，不过Insight最终还是调用GDB。在这里具体调试就不细说明了。 五：多平台代码移植后的代码(本文于2003年完成. 如需要转载 请联系jackforce@163.com)如果需要多个平台上面运行，就要在lib目录里面大做文章了。提供自己的函数库，并根据各个平台进行调整。Tar的代码由Config.h和一些编译选项来控制如何在各个不同的平台上面做编译。Lib则提供了很多C Library函数或者不同平台下面的其他函数的替代版本。这样Tar在编译过程中就不会因为某些平台下某些函数的缺失而编译不过。多平台支持，一般都是在代码中加上很多编译开关，在编译期间去分隔Linux，Windows或者其他平台下面的特殊代码。比如utime.h头文件的包含问题。因为文件在Linux（gcc）下面和Windows（cl）下所处的C Library目录不同。包含的处理办法就不一样。

可能需要这样写才能完全正确的包含。 `#if HAVE_UTIME_H`
`amp.#61663.-----如果是win32环境 # include amp.#61663.---- 如`
`果是Linux环境# include amp.#61663.--- 如果没有utime.h定义出`
`需要的结构 struct utimbuf{long actime.long modtime.}.#endif`在其他
其他的代码中基本上也是这样的处理。根据编译环境的不同来
编译不同的代码。这样的define的区隔，主要就是为了区隔不
同平台的不同细微区别。有的区别也许是某些常量没有定义
，有些区别是某些函数不存在。如果代码中调用函数在某些
平台下面不存在，就需要提供一个lib去提供这些函数。Tar
的Lib的作用也是如此。基本上代码的移植是前难后易。前期
首先要保证源代码本身的逻辑不能变动，所以在修改代码方
面只能尽量修改外围的代码，而不是修改源代码本身。如
果link过了之后，则就是一般的Windows下面的编程了，可
以根据需求任意修改移植后的代码了。最难的地方可能就是OS
级不同概念的替换了。C Library虽然在各个平台上有不同之
处，但是总是比较接近，不同的地方可以提供自己编写的代
码来替换。但是OS级的概念，和平台相关性太大，一般不太
容易替换。 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