

As is depicted in the picture above, my app randomly crashes with Fatal signal 11, code 1.

I searched it on Google, here is some information.

在POSIX兼容的平台上，SIGSEGV是当一个进程执行了一个无效的内存引用，或发生段错误时发送给它的信号。SIGSEGV的符号常量在头文件signal.h中定义。

*SIG*是信号名的通用前缀。*SEGV*是*segmentationviolation*（段违例）的缩写。

用Google获取这样一些知识。有多种可能导致SIGBUS信号:  
  
1) 硬件故障，不用说，程序员最常碰上的肯定不是这种情形。  
  
2) [**Linux**](http://lib.csdn.net/base/linux)平台上执行malloc()，如果没有足够的RAM，Linux不是让malloc()失败返回，  
   而是向当前进程分发SIGBUS信号。  
  
   注: 对该点执怀疑态度，有机会可自行测试确认当前系统反应。  
  
3) 某些[**架构**](http://lib.csdn.net/base/architecture)上访问数据时有对齐的要求，比如只能从4字节边界上读取一个4字节的  
   数据类型。IA-32架构没有硬性要求对齐，尽管未对齐的访问降低执行效率。另外  
   一些架构，比如SPARC、m68k，要求对齐访问，否则向当前进程分发SIGBUS信号。  
  
SIGBUS与SIGSEGV信号一样，可以正常捕获。SIGBUS的缺省行为是终止当前进程并产  
生core dump。  
  
SIGBUS与SIGSEGV信号的一般区别如下:  
  
1) SIGBUS(Bus error)意味着指针所对应的地址是有效地址，但总线不能正常使用该  
   指针。通常是未对齐的数据访问所致。  
  
2) SIGSEGV(Segment fault)意味着指针所对应的地址是无效地址，没有物理内存对  
   应该地址。