关于处理存储管理器每个操作可能存在的各种异常，首先考虑所有的操作，尤其是需要用户输入的操作，因为“一切输入都是有害的”，其次判断各种操作可能出现的异常，然后对这些异常进行分类，建立合适的异常类型，最后应用于程序中的各个操作模块。下面展开讨论。

首先，考虑所有的操作，基本可按时序分为两个阶段：第一阶段——用户输入命令，由命令解析器Parser进行分析；第二阶段——Parser将解析后的指令传递给存储管理器MemManager执行。下面考虑两个阶段的操作中可能出现的异常情况：

其次，根据以上的列举结果进行归纳分析，将异常情况分成三个大类：空操作对象、参数错误、空间不足，再分为若干个子类型，具体分类情况如下：

最后，在将上述异常类型应用于程序中的各个操作模块时，根据的原则是：模块负责什么，就检测哪部分的异常。例如，命令解析器（Parser）负责解析用户输入的命令，故Parser负责判断用户输入的命令是否合规，如果用户在没有初始化存储池的情况下输入“new a=20”，那么Parser任然认为这则命令是正确的，至于存储池是否已经初始化，则交给存储管理器（MemManager）进行判断，如果未初始化，则由MemManager抛出未初始化存储池异常（MemoryPoolUninitializedException）。再例如，如果用户已经初始化存储池后输入“new a=20”，那么存储管理器（MemManager）就照此执行，至于存储池的剩余空间是否足够分配给变量a，MemManager不予理睬，交给存储池分配器（Allocator）进行判断，如果存储池的剩余空间不足，则由Allocator抛出存储池空间不足异常（InsufficientMemoryPoolException）。