1. weakref\_impl对象会随着目标对象的生成而产生，但不一定会随着目标对象的释放而释放。例如：如果目标对象被

1个sp引用，但是同时被2个wp引用，那么在sp被删除的时候，删除了目标对象，但没有删除weakref\_impl对象，

只有在最后一个wp释放时，weakref\_impl对象会被释放。

2. 一个目标对象被多个sp指针引用，没有wp引用的情况下。释放这些sp的时候，delete会调用sp析构函数，

然后调用RefBase类的成员函数decStrong(), 最后一个sp被释放时，weakref\_impl对象数据成员mStrong会

从1减到0(注意mStrong的初始化值为1<<28, 从这个值可以判断出该目标对象有没有被sp指针引用过)，

同时释放目标对象。

3. 一个目标对象被多个wp指针引用，没有sp引用的情况下。delete这些wp的时候，会调用wp的析构函数，该函数会

调用函数decWeak()。当删除最后一个wp的时候，代码中只是删除了目标对象，而没有释放weakref\_impl对象，

暂时没发现在哪里释放了它。

4. 一个目标对象既有sp，又有wp来引用。如果sp先被删除光，那么最后一个sp删除的时候会释放掉目标对象，那么此时

mStrong = 0。在后续最后一个wp的释放过程中，在decWeak()函数中就会判断出impl->mStrong !=

INITIAL\_STRONG\_VALUE,而释放掉剩下的weakref\_impl对象了。如果先所以的wp删除光，此时mWeak还等于剩余的sp

的个数，所以此时的释放情况，同第2小点的说明。

5. 从wp定义来看，wp是不能直接操作对象的，必须先升级为sp才行。这个升级的过程是依靠函数promote()来完成的。

升级成功，返回新生成的sp对象指针，升级失败，返回NULL。需要注意的是，如果目标对象之前有过sp指向，但后来

将所有的sp释放完之后，此时目标对象是不存在的，那么此时用户还想将指向该目标对象的wp升级为sp的话，

此时就返回NULL。那么这个时候我们应该delete这些剩下的wp。