

手工声明事务或持久性数据/代码

```
1 void TMhashtable_update(TM_ARGDECL hashtable_t* hashtablePtr,  
2                          void* keyPtr, void* dataPtr) {  
3     list_t* chainPtr = get_chain(hashtablePtr, keyPtr);  
4     pair_t* pairPtr;  
5     pair_t updatePair;  
6     updatePair.firstPtr = keyPtr;  
7     pairPtr = (pair_t*)TMLIST_FIND(chainPtr, &updatePair);  
8     pairPtr->secondPtr = dataPtr;  
9 }
```

要求第三方库
支持事务性接口

禁止的操作，
导致运行时错误

(a)

无需更改的语法和语义

```
1 void hashtable_update(hashtable_t* hashtablePtr,  
2                       void* keyPtr, void* dataPtr) {  
3     list_t* chainPtr = get_chain(hashtablePtr, keyPtr);  
4     pair_t* pairPtr;  
5     pair_t updatePair;  
6     updatePair.firstPtr = keyPtr;  
7     pairPtr = (pair_t*)list_find(chainPtr, &updatePair);  
8     pairPtr->secondPtr = dataPtr;  
9 }
```

有效的操作，持久性内存系统
保证故障时数据一致性

(b)