

목표

1. 락프리 구조로 Thread safe 자료구조 작성
2. 락 기반 자료구조보다 빠른 속도

구현 스펙

1. Node를 이용한 리스트 기반의 큐
2. 메모리 풀 사용.
→ 락프리 큐 구조 특성상, 이미 반환된 메모리에 접근하는 경우가 있기 때문에, 전용 메모리 풀 필요
3. Double Compare and set와 유니크한 Count값을 이용
→ ABA문제 방지
4. Head는 무조건 Dummy.
5. 실제 데이터는 head->next부터.
→ head가 null이 되는 상황 방지.

LockFree-Queue 기본 구조

Class CLockFreeQueue<DATA>

Class CMemoryPool<stNode>

메모리 풀
(Node를 다룬다.)

HAS-A 관계

: stNode_Point는 stNode를 가지고 있다.

Struct stNode_Point

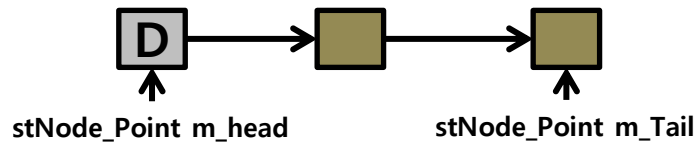
stNode* pPoint;
LONG64 Count;



Struct stNode

DATA data;
stNode* pNext;

내부 리스트



Enqueue(DATA data)

LockFree-Queue 내부 리스트에 노드 추가

Dequeue(DATA* pData)

LockFree-Queue 내부 리스트의 노드 1개 반환