

## OrdIntList.h

---

```
#ifndef __ORDINTLIST_H
#define __ORDINTLIST_H

#include "IntList.h"

class COrdIntList : public CIntList
{
public:
    COrdIntList(){};
    ~COrdIntList(){};

    bool insert( int iValue );
};

#endif
```

## OrdIntList.cpp

---

```
#include "OrdIntList.h"

bool COrdIntList::insert( int iValue )
{
    IntNode * pNode = new IntNode;

    if( pNode == NULL ) return false;

    pNode->m_iValue = iValue;

    // se lista vazia => inserir unico elemento
    if ( is_empty() )
    {
        pNode->m_pNext = NULL;

        m_pHead = pNode;
    }
    // encontrar local correcto para inserir
    else
    {
        // novo elemento e' menor que o inicio => inserir no inicio
        if ( iValue < m_pHead->m_iValue )
        {
            pNode->m_pNext = m_pHead;

            m_pHead = pNode;
        }
        // inserir no meio ou fim
        else
        {
            IntNode * pAnt = m_pHead;

            for ( IntNode * p = m_pHead->m_pNext; p != NULL && iValue >
p->m_iValue; p = p->m_pNext )
                pAnt = p;

            // pAnt aponta para o elemento onde deve ser feita a insercao
            pNode->m_pNext = pAnt->m_pNext;

            pAnt->m_pNext = pNode;
        }
    }

    m_iCount++;

    return true;
}
```