template<class List\_entry>

List<List\_entry>::List()

{

count = 0;

}

template<class List\_entry>

bool List<List\_entry>::empty()const

{

count == 0;

}

template<class List\_entry>

void List<List\_entry>::clear()

{

count = 0;

}

template<class List\_entry>

Error\_code List<List\_entry>:: retrieve(int position, List\_entry& x)const

{

if (empty())

return underflow;

if (position<0 || position>count)

return range\_error;

x = entry[position];

return x;

}

template<class List\_entry>

Error\_code List<List\_entry>::replace(int position, const List\_entry& x)

{

if (empty())

return underflow;

if (position<0 || position>count)

return range\_error;

entry[position] = x;

return success;

}

template<class List\_entry>

Error\_code List<List\_entry>::remove(int position, List\_entry& x);

{

if (empty())

return underflow;

if (position<0 || position>count)

return range\_error;

for (int i = position; i <= count; i++)

entry[i] = entry[i + 1];

count--;

return success;

}

问题：1.没有系统的知识框架

2.感觉第一二节课讲的比较抽象