作业 4 参考答案（by 宋天恩）

P124-3

```
template<class T>
void chain<T>::set(int theIndex, const T &theElement) {
	checkIndex(theIndex);
	chainNode<T> *currentNode = firstnode;
	for (int i = 0; i < theIndex; i++)
		currentNode = currentNode->next;
	currentNode->element = theElement;
}
//时间复杂度为 O(listSize)
```

P125-15

```
template<class T>
void chain<T>::reverse() {
	if (empty())
		return;
	chainNode<T> *ptr1 = firstnode, *ptr2 = firstnode->next, *record;
	firstnode->next = NULL;
	while (ptr2 != NULL)
	{
		record = ptr1;
		ptr1 = ptr2;
		ptr2 = ptr2->next;
		ptr1->next = record;
	}
	firstnode = ptr1;
}
//时间复杂度为 O(listSize)
```

P125-17

```
//类定义时需要将 meld 声明为友元函数
template<class T>
extendedChain meld(const extendedChain<T> &a, const extendedChain<T> &b) {
	if (a.empty())
		return b;
	extendedChain<T> c;
	chainNode<T> *p = a.firstnode, *q = b.firstnode, *k;
	c.firstnode = new chainNode<T>(p->element);
	k = c.firstnode;
```

```
        p = p->next;
        while (p != NULL && q != NULL)
        {
            k->next = new chainNode<T>(q->element);
            q = q->next;
            k = k->next;
            k->next = new chainNode<T>(p->element);
            p = p->next;
            k = k->next;
        }
        while (p != NULL)
        {
            k->next = new chainNode<T>(p->element);
            p = p->next;
            k = k->next;
        }
        while (q != NULL)
        {
            k->next = new chainNode<T>(q->element);
            q = q->next;
            k = k->next;
        }
        return c;
}
//时间复杂度为 O(a.listSize＋b.listSize)，证明略
```