3.列表

无序列表:插入

邓俊辉

deng@tsinghua.edu.cn

```
❖ template <typename T> Posi(T) List<T>::insertB( Posi(T) p, T const & e )
   { __size++; return p-><u>insertAsPred(</u> e ); } //e当作p的前驱插入(Before)
❖ template <typename T> //前插入算法(后插入算法完全对称)
   Posi(T) <u>ListNode</u><T>::<u>insertAsPred(</u> T const & e ) { //O(1)
      Posi(T) x = new <u>ListNode(</u> e, pred, this ); //创建(耗时 100 倍)
      pred->succ = x; pred = x; return x; //建立链接,返回新节点的位置
                    (a)
                                                                (d)
                                                                                   SUCC
             Succ
                                this
                                                                             this
                                                                new
pred
                          pred
                    (b)
                                                                (c)
                                                         succ
             SUCC
                                       SUCC
                                                                                   SUCC
                                this
                                                                             this
pred
                                            pred
                                                                      pred
                          pred
                                                                      succ
                          succ
                                                                new
                    new
             pred
                                                     Data Structures & Algorithms, Tsinghua University
```