第二章 作业

姓名： 崔冠宇 学号： 2018202147

1. 基础知识题

习题2.1

答：所谓首元节点，是指存储线性表第一个元素的节点结构体。而为了在表的头部添加、删除节点等操作方便，以及为了统一空表和非空表，而虚设了一个不储存内容的节点，称为头节点。头指针是指指向头节点的指针，用它来“统领”链表。

习题2.2

1. , 为表的元素个数 （2）一定 不一定
2. 上一节点的指针域 （4）为了方便统一空表、非空表的操作

习题2.3

答：由于顺序表具有随机存储的优点，而插入、删除的时间复杂度相比而言较高，所以在需要频繁存取的操作，同时又不太需要插入删除的场合，选择顺序表比链表好。

习题2.4

（1）

2

5

7

3

8

6

4

^

…

L

P

Q

R

S

（2）

2

5

7

3

8

6

4

^

…

L

Q

R

P

S

（3）

2

5

7

5

8

6

4

^

…

L

P

Q

R

S

（4）

2

5

7

7

8

6

4

^

…

L

P

Q

R

S

（5）

2

5

7

3

5

6

4

^

…

L

P

Q

R

S

（6）

2

10

14

6

16

12

8

^

…

L

P

Q

R

S

（7）

2

10

14

6

16

12

4

^

…

L

P

Q

R

S

习题2.5

L

P

L

1

3

5

7

^

P

L

1

2

3

8

^

4

5

6

7

P

4

6

L

2

P

7

8

^

习题2.6

a:（4）（1） b:（7）（11）（8）（4）（1）

c:（5）（12） d:（9）（6）（1）

习题2.7

a:（11）（3）（14） b:（10）（12）（8）（11）（3）（14）

c:（10）（12）（7）（3）（14） d:（12）（11）（3）（14）

e:（12）（9）（11）（3）（14）

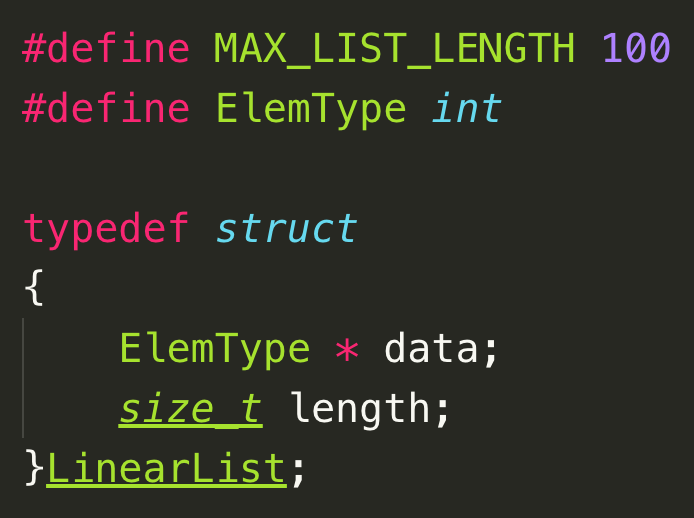
二、算法题

要求：

* 所有的题目都给出存储结构的定义、算法的思想说明及算法复杂度的分析
* 下面各算法题要求纸面提交，但要求学生在上机时一定要把提交的算法实现并测试

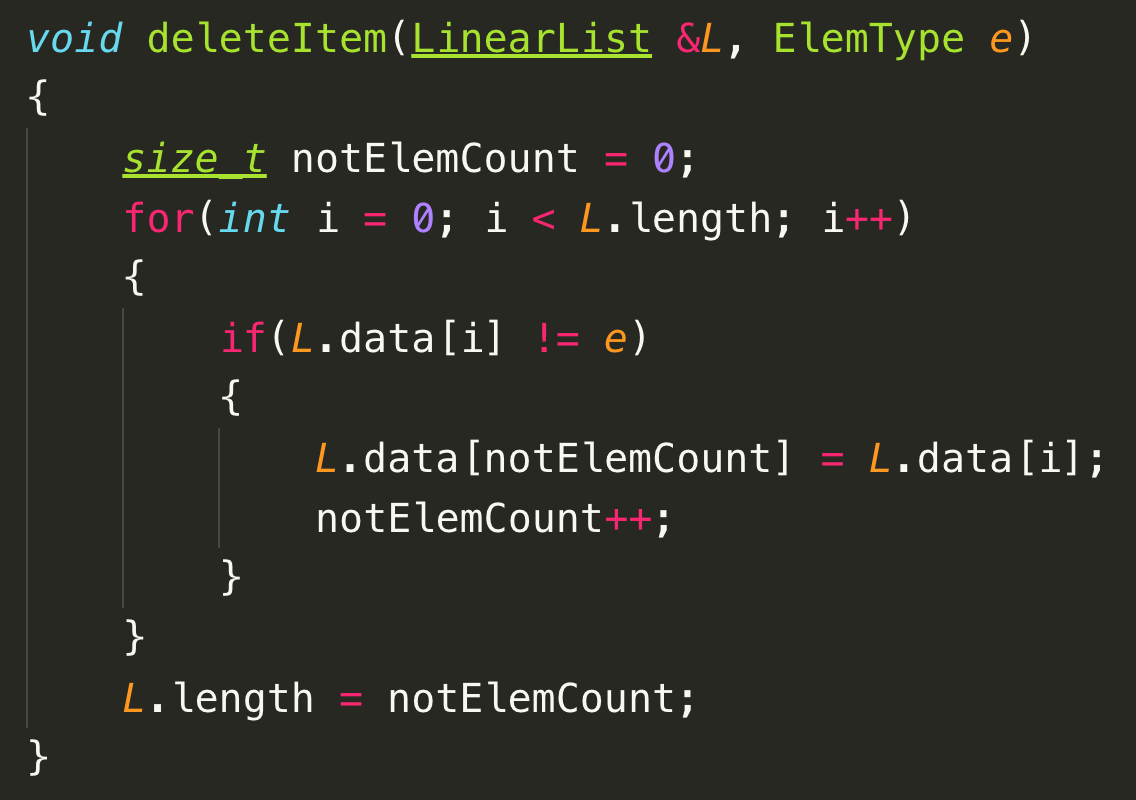
1. 已知长度为n的线性表A采用**顺序存储**结构，请写一时间复杂度为0(n)、空间复杂度为0(1)的算法，该算法删除线性表中所有值为item的数据元素。（O（1）表示算法的辅助空间为常量）。

存储结构定义:



算法如下：

基本思想是在遍历时记录下非item元素的个数，并将非item元素向前挪动。

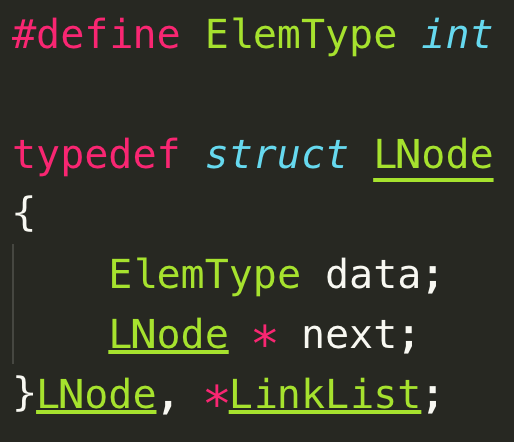


复杂度分析：

由于只遍历了一遍，故时间复杂度为。

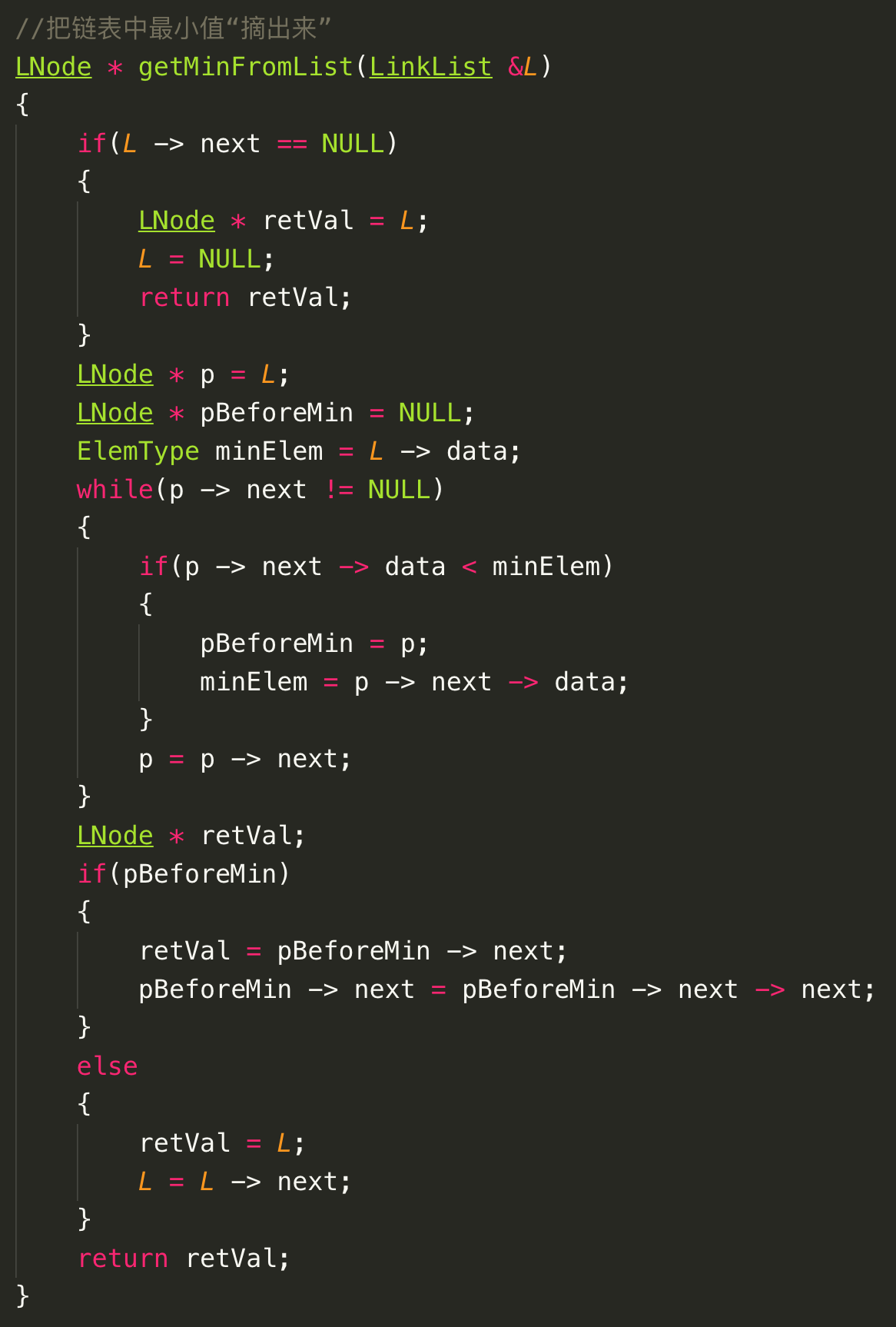
1. 已知不带头结点的线性链表list，链表中结点构造为（data、link），其中data为数据域，link为指针域。请写一算法，将该链表按结点数据域的值的大小从小到大重新链接。要求链接过程中不得使用除该链表以外的任何链结点空间。

存储结构定义如下：



算法如下：

基本思想是先找到全链表最小值作为头，然后每次遍历将剩余部分最小结点取出加到最后。





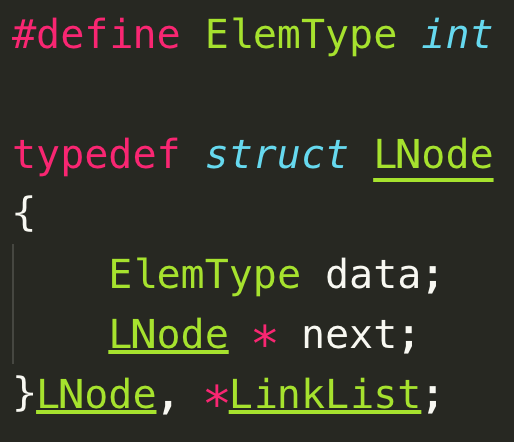
复杂度分析：

设原链表长度为,由于每次遍历减少一个节点，故总共需要访问节点，所以时间复杂度为.

1. 试编写在带头结点的单链表中删除（一个）最小值结点的（高效）算法。

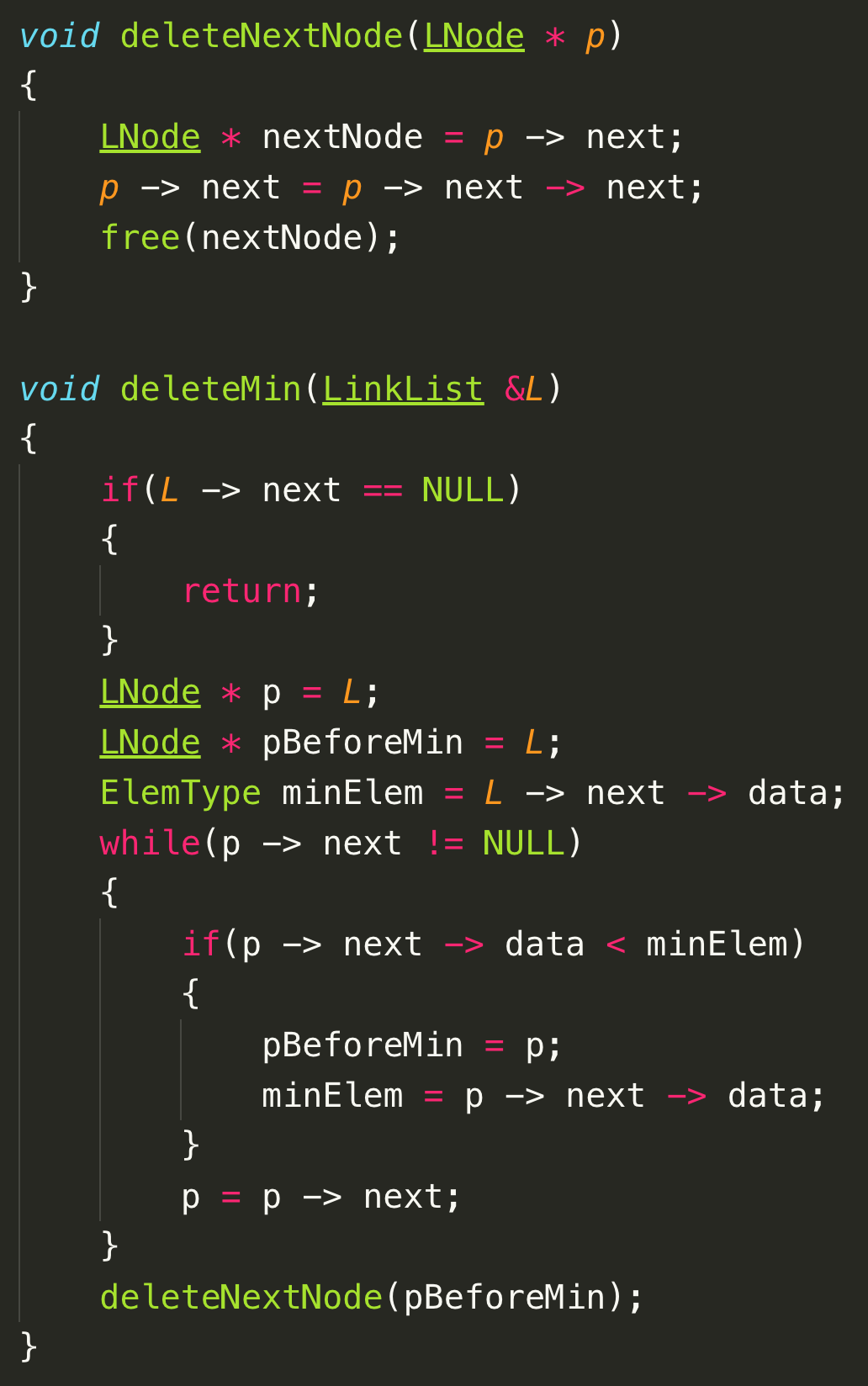
void delete（Linklist &L）

存储结构定义如下：



算法如下：

基本思想是遍历链表，实时记录最小值和最小值之前节点的地址，最后进行删除。



复杂度分析：

因为只遍历一次，时间复杂度为。