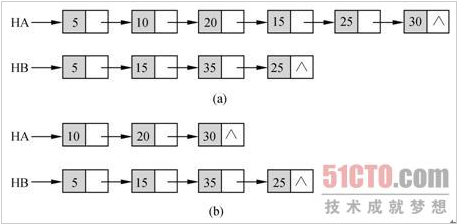
**试题五（共15分）**

**阅读下列说明和C函数，将应填入 （n） 处的字句写在答题纸的对应栏内。**

**说明**

**已知集合A和B的元素分别用不含头结点的单链表存储，函数Difference()用于求解集合A与B的差集，并将结果保存在集合A的单链表中。例如，若集合A=5，10，20，15，25，30，集合B=5，15，35，25，如图（a）所示，运算完成后的结果如图（b）所示。**

****

**链表结点的结构类型定义如下：**

**typedef struct Node{**

**ElemType elem;**

**structNode\*next;**

**}NodeType;**

**C函数**

**void Difference(NodeType \*\*LA, NodeType \*LB){**

**NodeType \*pa,\*pb,\*pre, \*q;pre = NULL;**

**（1） ;**

**while (pa) {**

**pb = LB;**

**while ( （2） )**

**pb=pb->next;**

**if ( （3） ){**

**if (!pre) \*LA = （4） ;**

**else（5） =pa->next;**

**q= pa;**

**pa = pa->next;**

**free(q);**

**}**

**else{**

**（6） ;**

**pa =pa->next;**

**}**

**}**

**}**

**试题五分析**

**本题考查链表结构上的基本运算。**

**集合A与B的差是指在集合A中而不在集合B中的元素。本题用链表表示集合并将运算结果用表示集合A的链表存储，因此涉及到链表上的查找、删除基本运算。**

**基本思路为：对于集合A中的每个元素，在集合B中进行查找，若找到，则应将该元素从集合A中去掉；否则保留，用两层循环实现，外层循环用于遍历集合A，内层循环遍历集合B。**

**代码中的指针pa用于指向集合A的元素；pb指向集合B的元素；临时指针q指向需要被删除的元素；pre用于实现删除时结点的链接，与pa保持所指结点的前后继关系。**

**显然，pa需要一个初始值，即指向集合A的第一个元素结点。由于参数LA是指向集合A第一个结点的指针的指针，因此空（1）处应填入pa=\*LA。**

**在内层循环中遍历集合B时，初始时令pb指向B的第一个元素（pb=LB），此后应在链表中查找与A中当前元素相同者，因此空（2）处应填入pb && pb->elem!=pa->elem。**

**此后，应判断在B中是否找到指定元素。显然，若找到（即pb->elempa->elem），则指针pb不为空，否则，pb为空。因此，空（3）处填入pb或pb!=NULL，空（6）处则填入pre=pa。**

**由于链表不带头结点，因此，当需要删除集合A的第一个元素时，表示该集合的链表头指针会被修改。pre初始值为NULL，可标志删除的是否为A的第一个元素。因此查找成功时，pre为空（!pre成立）表示需要删除A的第一个元素（pa指针所指），使得A的头指针指向第二个元素，即应将\*LA更新为pa->next，空（4）处填入pa->next。如果删除的不是第一个元素，则由于pa指向被删除的元素，而且pre与pa所指元素保持前后继关系，因此空（5）处应填入pre->next。**

**参考答案**

**（1）pa = \*LA （2）pb && pb->elem !=pa->elem，或其等价表示**

**（3）pb或pb!=NULL （4）pa->next，或(\*pa). next，或其等价表示**

**（5）pre->next，或(\*pre). next （6）pre = pa**