## 题目描述：

## 输入一个复杂链表（每个节点中有节点值，以及两个指针，一个指向下一个节点，另一个特殊指针指向任意一个节点），返回结果为复制后复杂链表的head。（注意，输出结果中请不要返回参数中的节点引用，否则判题程序会直接返回空）

## 解题思路：

 1、复制每个节点，如：复制节点A得到A1，将A1插入节点A后面

 2、遍历链表，A1->random = A->random->next;

 3、将链表拆分成原链表和复制后的链表

## 算法描述：

/\*

struct RandomListNode {

int label;

struct RandomListNode \*next, \*random;

RandomListNode(int x) :

label(x), next(NULL), random(NULL) {

}

};

\*/

class Solution {

public:

RandomListNode\* Clone(RandomListNode\* pHead)

{

if (pHead == NULL)

return NULL;

RandomListNode \*CurrNode = pHead;

while (CurrNode)

{

RandomListNode \* Node = new RandomListNode(CurrNode->label);

Node->next = CurrNode->next;

CurrNode->next = Node;

CurrNode = Node->next;

}

CurrNode = pHead;

while(CurrNode)

{

RandomListNode \* Node = CurrNode->next;

if (CurrNode->random)

{

Node->random = CurrNode->random->next;

}

CurrNode = Node->next;

}

RandomListNode \* cpHead = pHead->next;

RandomListNode \* p = NULL;

CurrNode = pHead;

while(CurrNode->next)

{

p = CurrNode->next;

CurrNode->next = p->next;

CurrNode = p;

}

return cpHead;

}

};