138. Copy List with Random Pointer

题目描述：https://leetcode.com/problems/copy-list-with-random-pointer/

> 深拷贝一个具有random节点的链表

解题思路：

注意通过next不一定访问全

代码：

```
/**
 * Definition for singly-linked list with a random pointer.
 * struct RandomListNode {
 *     int label;
 *     RandomListNode *next, *random;
 *     RandomListNode(int x) : label(x), next(NULL), random(NULL) {}
 * };
 */
class Solution {
public:
    RandomListNode *copyRandomList(RandomListNode *head) {
        if(head == NULL) return NULL;
        // if(head->next == NULL) return new RandomListNode(head->label);
        unordered_map<RandomListNode*, RandomListNode*> mp;
        RandomListNode* fakeHead = new RandomListNode(0);
        RandomListNode* pre = fakeHead;
        while(head) {
            if(mp.find(head) == mp.end()) {
                RandomListNode* cur = new RandomListNode(head->label);
                mp.insert(pair<RandomListNode*, RandomListNode*>(head, cur));
            }
            pre->next = mp[head];
            if(head->random != NULL && mp.find(head->random) == mp.end()) {
                RandomListNode* ran = new RandomListNode(head->random->label);
                mp.insert(pair<RandomListNode*, RandomListNode*>(head->random, ran
));
            }
            mp[head]->random = mp[head->random];
            pre = mp[head];
            head = head->next;
        }
        return fakeHead->next;
    }
};
```