138. Copy List with Random Pointer ★

Question | Editorial Solution

My Submissions (/problems/copy-list-with-random-pointer/submissions/)

Total Accepted: **76145** Total Submissions: **290190** Difficulty: **Hard**

Notes

A linked list is given such that each node contains an additional random pointer which could point to any node in the list or null.

Return a deep copy of the list.

Subscribe (/subscribe/) to see which companies asked this question

Show Tags

Show Similar Problems

Have you met this question in a real interview? Yes No

Discuss (https://leetcode.com/discuss/questions/oj/copy-list-with-random-pointer)

Pick One (/problems/random-one-question/)

C++

```cpp
/**
 * 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) {
        unordered_map<RandomListNode*, RandomListNode *> hash;
        for (auto it = head; it != NULL; it = it->next) {
            hash[it] = new RandomListNode(it->label);
        }
        for (auto it = head; it != NULL; it = it->next) {
            hash[it]->next = hash[it->next];
            hash[it]->random = hash[it->random];
        }
        return hash[head];
    }
};
```

✉ Send Feedback (mailto:admin@leetcode.com?subject=Feedback)

Submit Solution

Notes

✉ Send Feedback (mailto:admin@leetcode.com?subject=Feedback)