589. N-ary Tree Preorder Traversal

Tags	range-based-loop
Property	@September 20, 2022

Question

原文：

Given the `root` of an n-ary tree, return *the preorder traversal of its nodes' values*.

Nary-Tree input serialization is represented in their level order traversal. Each group of children is separated by the null value (See examples)

我的理解：

給定一個 root ，return 前序遍歷後的全部 node value。

N-ary tree 以NULL做分割，先輸入根節點value 然後 NULL，接下來輸入的節點都是根節點的子節點直到出現NULL，接下來的輸入就根節點第一個子節點的子節點，直到出現NULL

翻譯：

给定一个nary树的`根'，返回*其节点值的前序遍历*。

Nary-Tree的输入序列化是以它们的级别顺序遍历来表示的。每一组子节点都由空值隔开（见示例）。

自評翻譯正確性：80%

- Word Memory：
  - serialization 序列化
  - represented 代表
  - separated 分離的

Code

```
/*
// Definition for a Node.
class Node {
public:
    int val;
    vector<Node*> children;

    Node() {}

    Node(int _val) {
        val = _val;
    }

    Node(int _val, vector<Node*> _children) {
        val = _val;
        children = _children;
    }
};
*/

class Solution {
public:
    vector<int> preorder(Node* root) {
        vector<int>ans;
        if(root){
            pre(root,&ans);
        }
        return ans;
    }
    void pre(Node* root,vector<int>* ans){
        ans->push_back(root->val);
        for(Node* child : root->children){
            pre(child,ans);
        }
    }
};
```

> 思路：（本題基於參考其他人做法）ans 拿來儲存 val 值，假如root本身不為空就進入pre function，本題解題重點在於Node* child : root->children，該寫法類似 python 裡的 for i in range(n.j)，child會依序變成：後的集合(這邊是root的子樹們)，依序遞迴下去就能遍歷完n-ary tree，（補充：child的部分，型別宣告要跟後面要遍歷的集合內的物件型別一樣，也可以將型別宣告成Auto，就會自動宣告成符合後面集合的型別）

Success Details ›

Runtime: 47 ms, faster than 31.67% of C++ online submissions for N-ary Tree Preorder Traversal.

Memory Usage: 11.3 MB, less than 27.80% of C++ online submissions for N-ary Tree Preorder Traversal.

Next challenges:

Binary Tree Preorder Traversal | N-ary Tree Level Order Traversal | N-ary Tree Postorder Traversal

Show off your acceptance:

Time Submitted	Status	Runtime	Memory	Language
09/20/2022 15:55	Accepted	47 ms	11.3 MB	cpp
09/20/2022 15:55	Accepted	57 ms	11.3 MB	cpp
09/20/2022 15:52	Accepted	51 ms	11.3 MB	cpp

優良code參考

```
```

> 思路：