

动态规划 3

树形背包问题

基本模型

给定一棵 n 阶树，每个结点拥有重量 w_i 和价值 v_i 且选取某个结点必须选取该结点的父结点。

给定背包容量 M 问可以得到的最大价值。

DFS 序优化

考虑后序遍历结点并编号，设 $dp(i, j)$ 表示仅考虑后序遍历前 i 个结点的容量为 j 的背包的最大价值。

考虑新加入的第 i 个结点，发现后序遍历前 $[i - sz_i + 1, i]$ 个结点代表结点 i 的子树。

分别考虑不选取第 i 个结点和选取第 i 个结点的情况，于是有状态转移方程

$$dp(i, j) = \max(dp(i - sz_i, j), dp(i - 1, j - w_i) + v_i)$$

```
void dfs(int u){
    sz[u]=1;
    for(int i=head[u];i;i=edge[i].next){
        int v=edge[i].to;
        dfs(v);
        sz[u]+=sz[v];
    }
    dfs_t++;
    _for(i,0,w[u])
    dp[dfs_t][i]=dp[dfs_t-sz[u]][i];
    _for(i,w[u],MAXM)
    dp[dfs_t][i]=max(dp[dfs_t-1][i-w[u]]+v[u],dp[dfs_t-sz[u]][i]);
}
```

From: <https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link: https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:legal_string:jxm2001:%E5%8A%A8%E6%80%81%E8%A7%84%E5%88%92_3&rev=1610593126

Last update: 2021/01/14 10:58