洛谷P1122

【算法分析】

题意是在一个无环无向图（树）中删去一些边及边上的顶点，使剩下的顶点上的值最大。所以可以用邻接表方式保存图形，用dfs从任一顶点搜索，遇到子树价值为负数的则剪去，否则累加即可。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51 | #include <bits/stdc++.h>  using namespace std;  const int inf =0x7fffffff;  using namespace std;  struct node  {  int to,next;  } e[32010]; //结构体记录每条边的终点和其相邻的边  int n,ans,k;  int f[16010],head[16010];  void Add(int u,int to)//建邻接表  {  e[++k].to=to;  e[k].next=head[u];  head[u]=k;  }  int dfs(int x,int father)  {  int sum=f[x];//每棵子树至少都要包含本身的值  for(int i=head[x]; i; i=e[i].next) //枚举第x节点的儿子  if(e[i].to!=father)//判断是否搜回到父亲节点,避免再搜回来  {  int p=dfs(e[i].to,x);//x节点的一棵子树的最大价值  sum+=max(0,p);//子树价值大于0不剪,累加  }  ans=max(ans,sum);//用ans记录每棵子树的价值中的max  return sum;  }  int main()  {  cin>>n;  for(int i=1; i<=n; i++)  {  cin>>f[i];  ans=max(ans,f[i]);//若最后只剩下一个节点，则该点为答案  }  for(int i=1; i<n; i++)  {  int a,b;  cin>>a>>b;  Add(a,b);//建正向边  Add(b,a);//建反向边  }  dfs(1,-1);//随便找个1点作根,根没有父亲所以可设为-1  cout<<ans<<endl;  return 0;  } |