

# **GHOST: Game Tree**

**A Small Example with  
a Picture and Some Code**

**TJHSST  
December 2008**

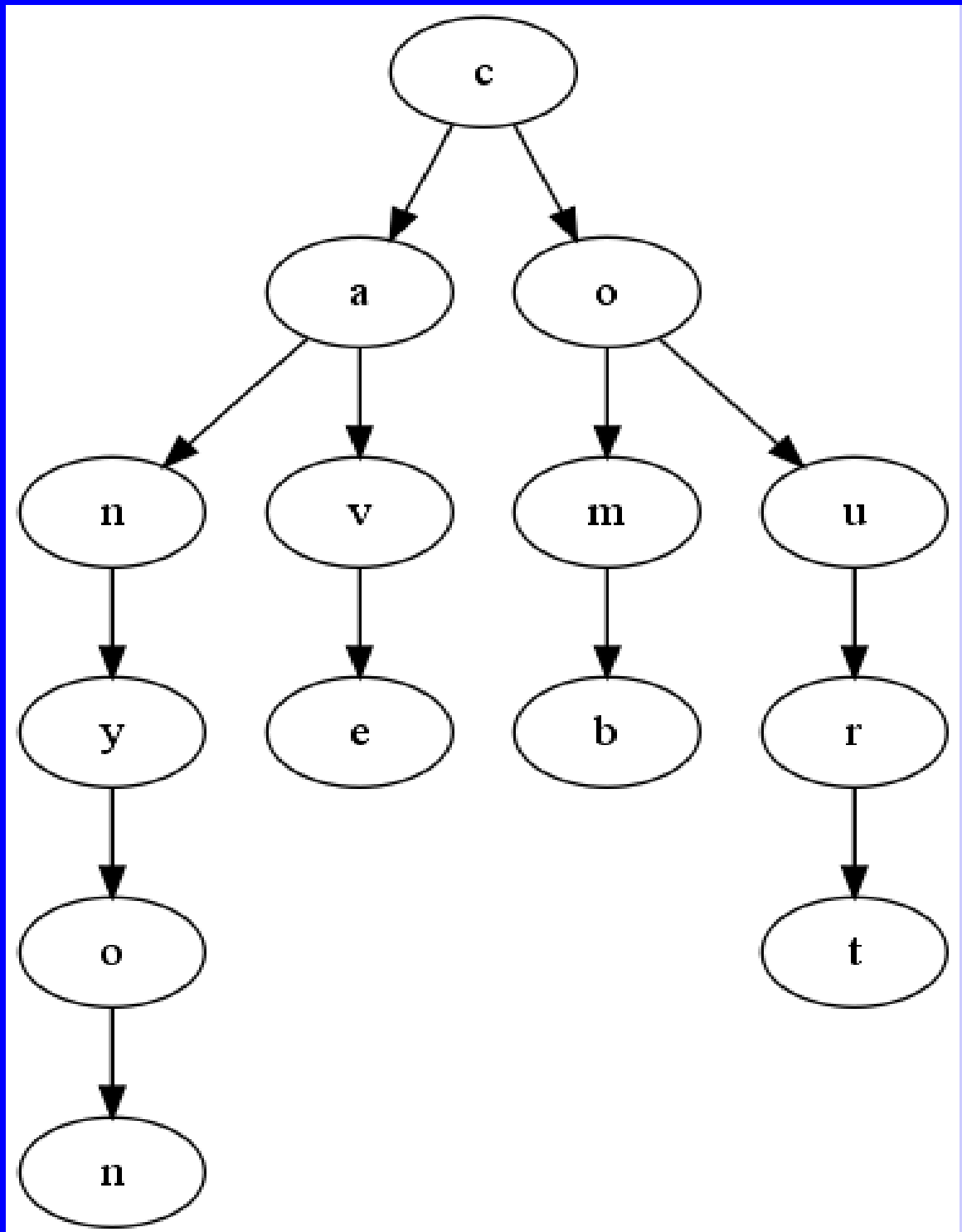
small.txt

canyon

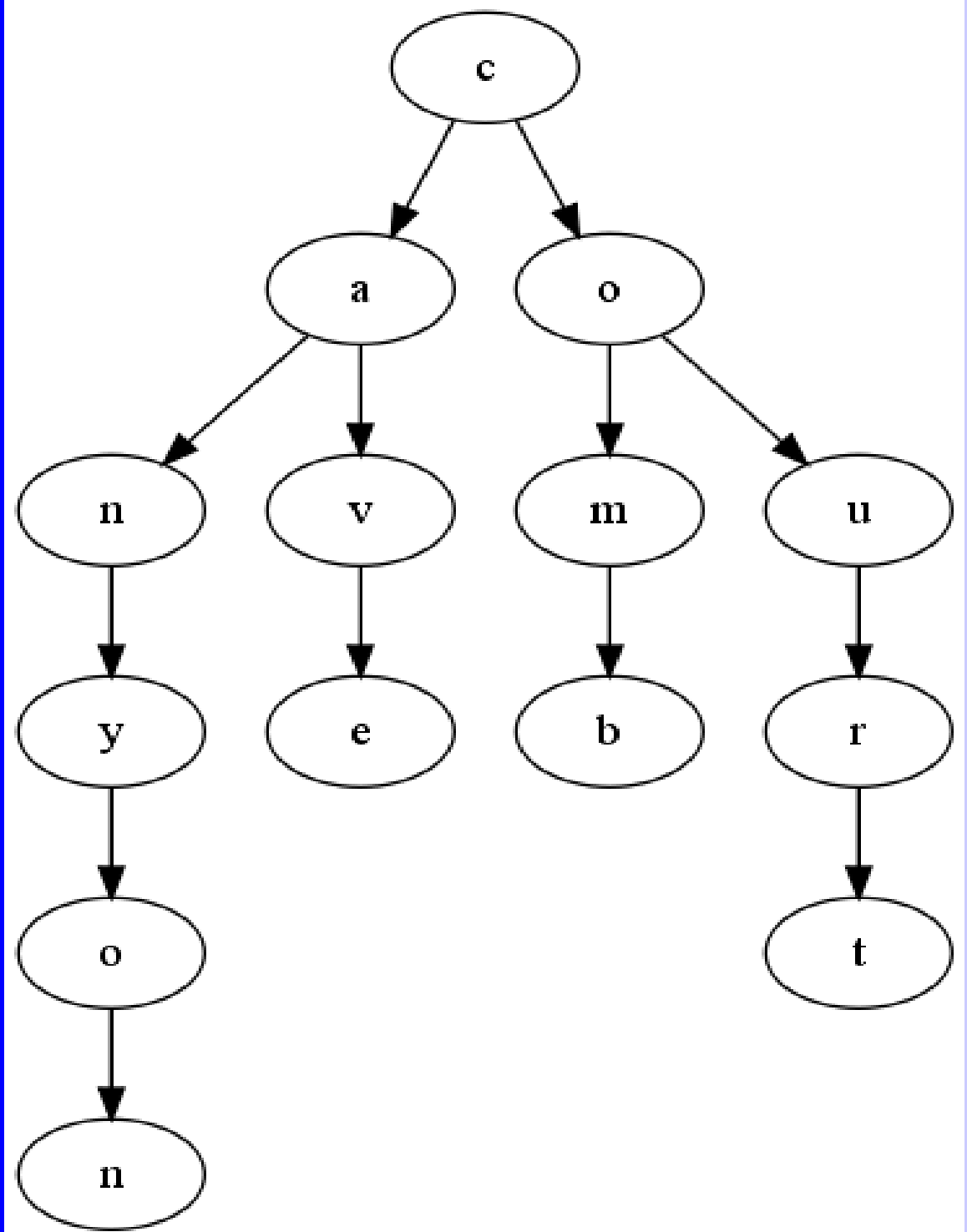
cave

comb

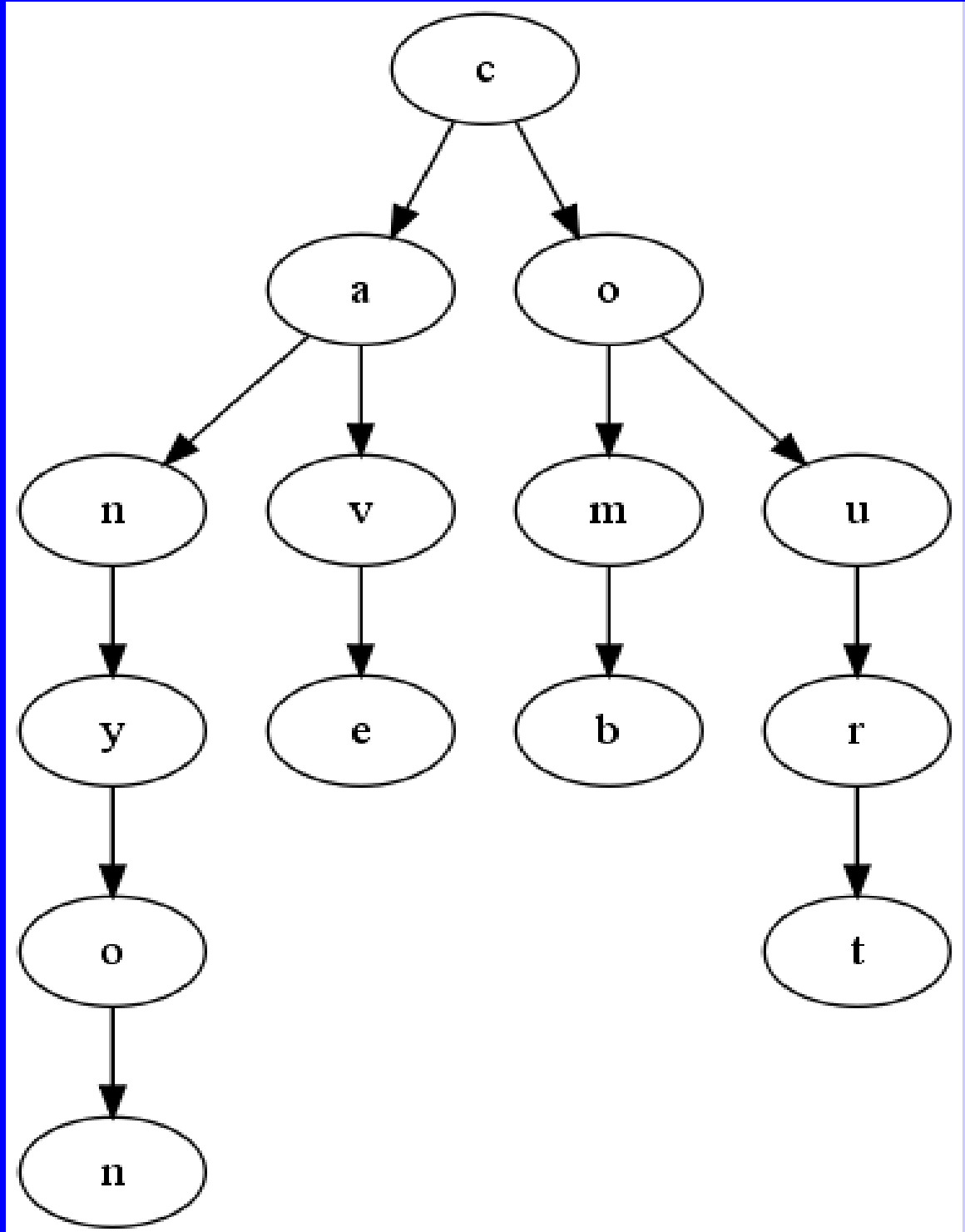
court



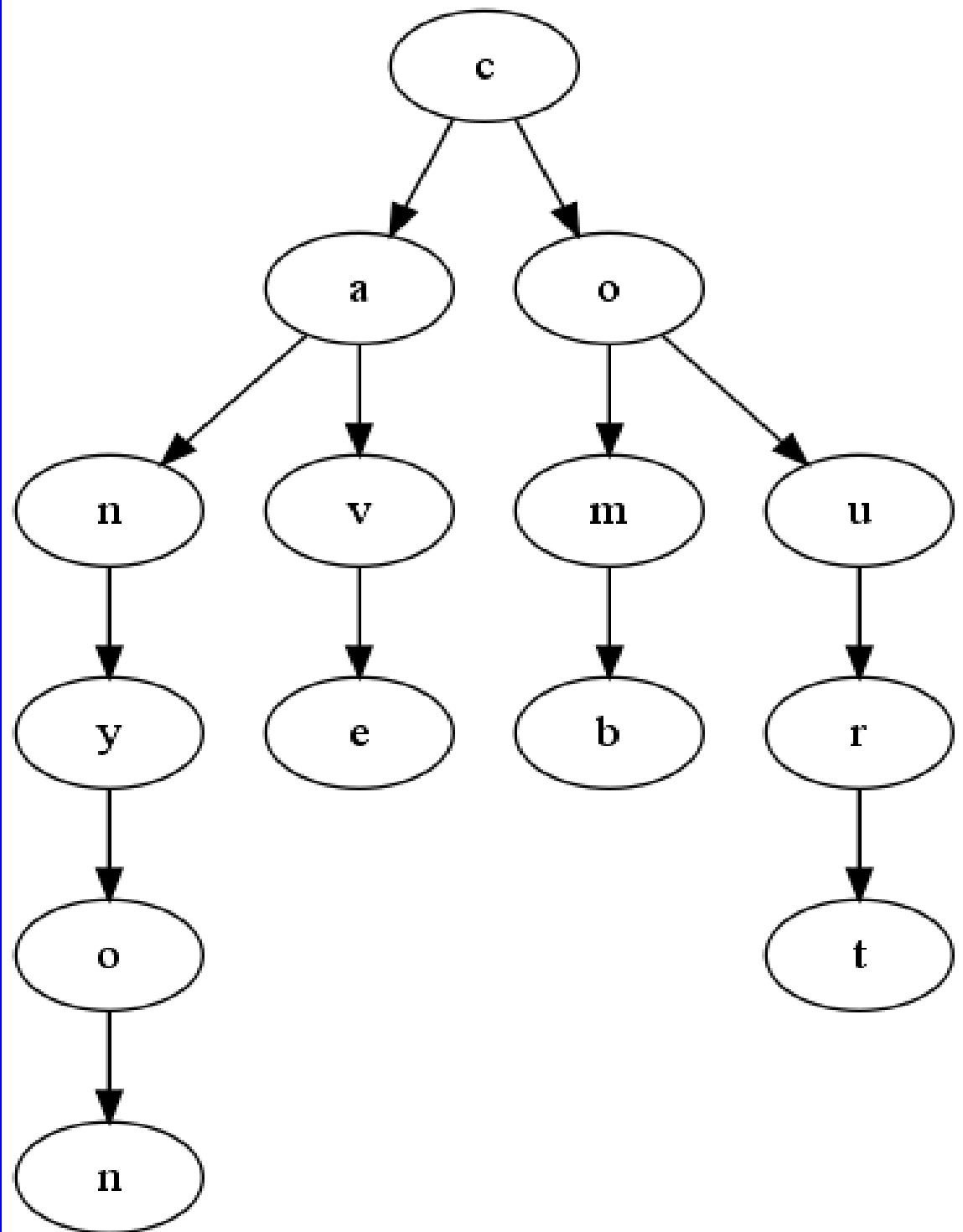
**Player One: C**  
**Player Two: O**  
**Player One: M**  
**Player Two: B**  
**Player One wins.**



**Player Two can't force the game towards COURT because Player One gets to take its own turn.**



**So, the presence of a favorable outcome down a particular branch is, *by itself*, not good enough to win.**



# Some Tree Code

**class Node:**

```
def __init__(self,val):  
    self.val=val  
    self.chldrn=[]
```

**def display\_tree(t,k):**

```
    print '\t'*k,t.val  
    for p in t.chldrn:  
        display_tree(p,k+1)
```

**In main:**

```
root=Node('*')
```

```
for k in a:
```

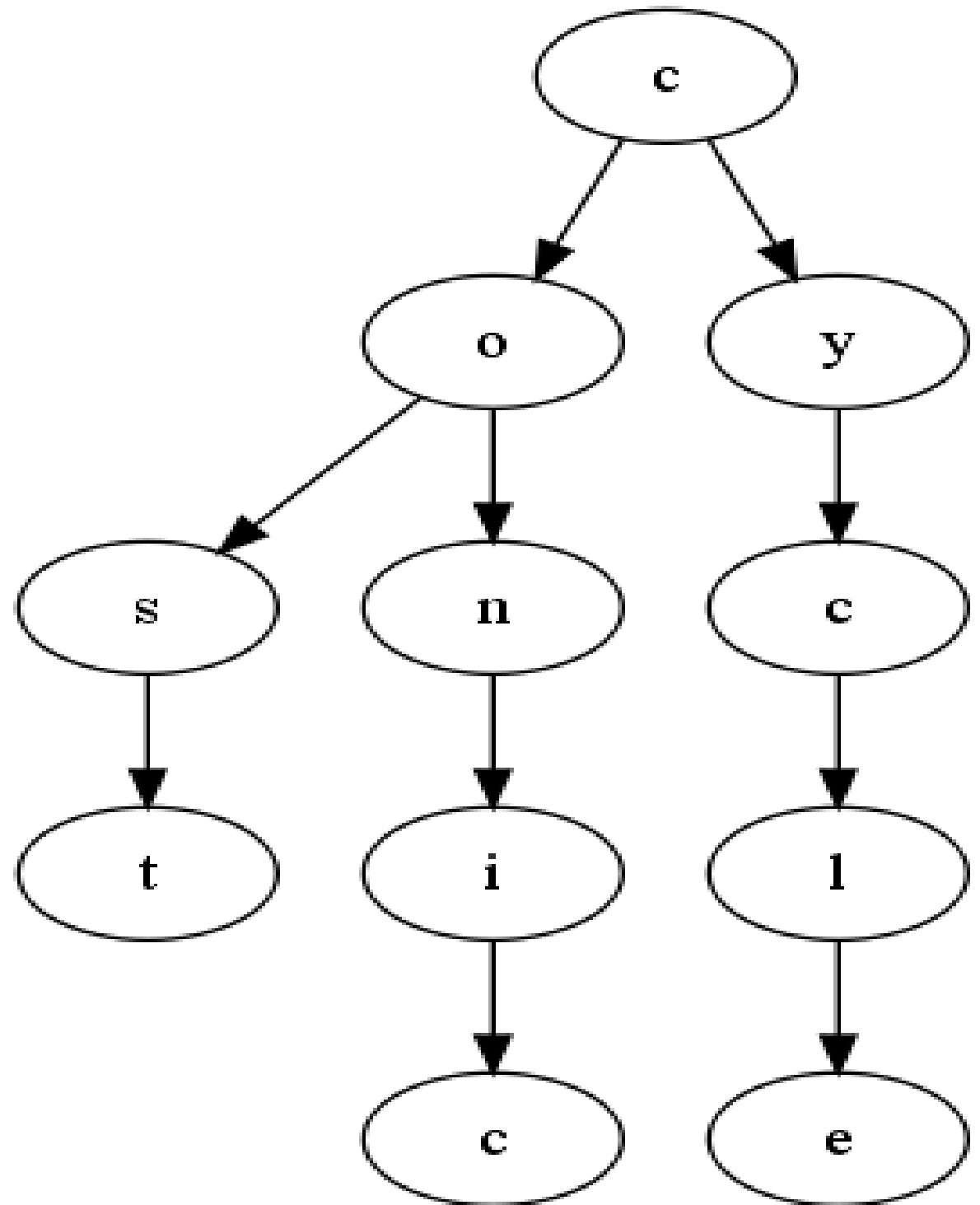
```
    print k
```

```
    # add k to tree
```

```
display_tree(root,0)
```

**cost, conic, cycle**

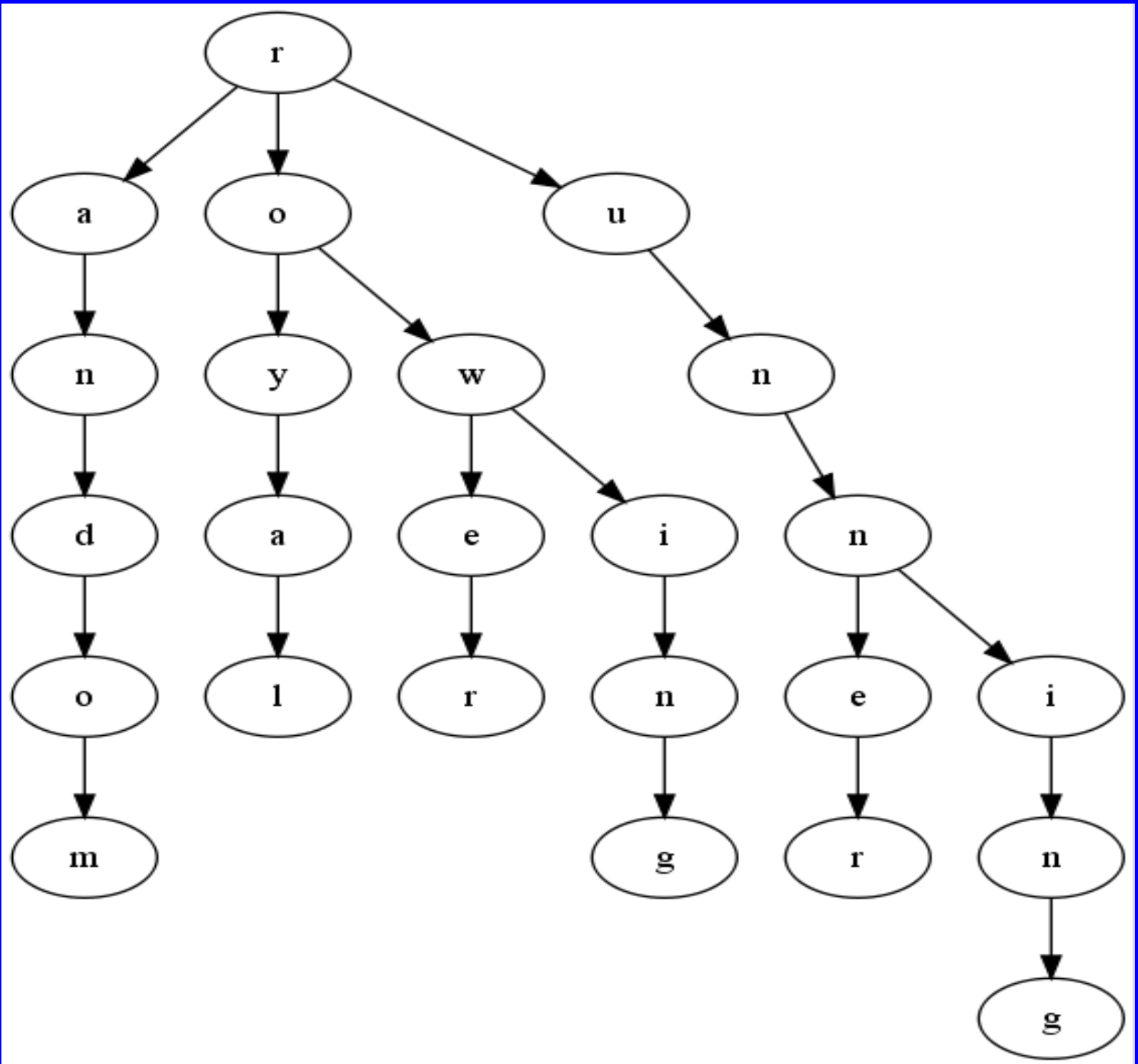
**Player Two can win, but not by going towards any favorable path it likes.**



# Keep in Mind

- **All of this is more complicated with three or four or five or six players**
- **Each player gets to make its own move**
- **Assume each player chooses moves in their own favor (i.e., no one is trying to lose)**
- **But, could include random mistakes and/or teaming up against a really good player**

random  
royal  
rower  
rowing  
runner  
running



random  
royal  
rower  
rowing  
runner  
running

