综合案例

下面的案例是一个众筹合约的案例，里面有两个角色，一个是投资人`Investor`，也就是`出资者`。另一个角色是被赞助者`BySponsor`，被赞助者。一个`Investor`可以给多个`BySponsor`赞助，一个`BySponsor`也可以被多个`Investor`赞助。

完整合约：

```
pragma solidity ^0.4.24;

contract CrowdFunding {
    //投资者
    struct Investor {
        //投资者地址
        address addr;
        //投资数量
        uint count;
    }

    //被赞助者
    struct BySponsor {
        address addr;
        uint goalCount;
        uint receiveCount;
        uint investorNum;
        mapping (uint=>Investor) investors;
    }

    uint bySponsorNum = 0;
    mapping (uint=>BySponsor) bySponsors;

    function getGoalCount(uint bySponsorId) constant returns (uint) {
        BySponsor memory bySponsor = bySponsors[bySponsorId];
        return bySponsor.goalCount;
    }
    function getReceiveCount(uint bySponsorId) constant returns (uint) {
        BySponsor memory bySponsor = bySponsors[bySponsorId];
```

```
        return bySponsor.receiveCount;
    }

    function newBySponsor() payable {
        bySponsorNum++;
        // 获取全局变量msg.value的方法必须使用payable标记。
        BySponsor memory bySponsor = BySponsor(msg.sender, msg.value, 0, 0);
        bySponsors[bySponsorNum] = bySponsor;
    }

    //投资者赞助被赞助者
    function sponsor(uint bySponsorId) payable {
        BySponsor storage bySponsor = bySponsors[bySponsorId];
        bySponsor.receiveCount += msg.value;
        bySponsor.investorNum++;
        bySponsor.investors[bySponsor.investorNum] = Investor(msg.sender,
msg.value);
        bySponsor.addr.transfer(msg.value);
    }

    function checkStandard(uint bySponsorId) constant returns (bool) {
        BySponsor bySponsor = bySponsors[bySponsorId];
        if (bySponsor.receiveCount >= bySponsor.goalCount && bySponsor.goalCount >
0) {
            return true;
        } else {
            return false;
        }
    }
}
```