# Demonstration of the Eisenberg/Noe Algorithm

We start by initializing a simple system of three nodes:

```
vecE = [10;10;10];
matL = [0 20 10;5 0 24;10 0 0];
```

We see that, before clearing, only the first node is in default:

```
vecPbar = sum(matL,2);
matPi = matL ./ repmat(vecPbar,1,length(vecE));
vecEquity = vecE + matPi'*vecPbar - vecPbar;

disp([{'Equity before clearing' 'Defaulted'};
    num2cell([vecEquity vecEquity < 0])])
```

```
    'Equity before clearing'    'Defaulted'
    [                    -5]    [        1]
    [                     1]    [        0]
    [                    34]    [        0]
```

We compute the clearing payment vector and see that all nodes face losses due to contagion, and that the second node is pushed into default through these losses:

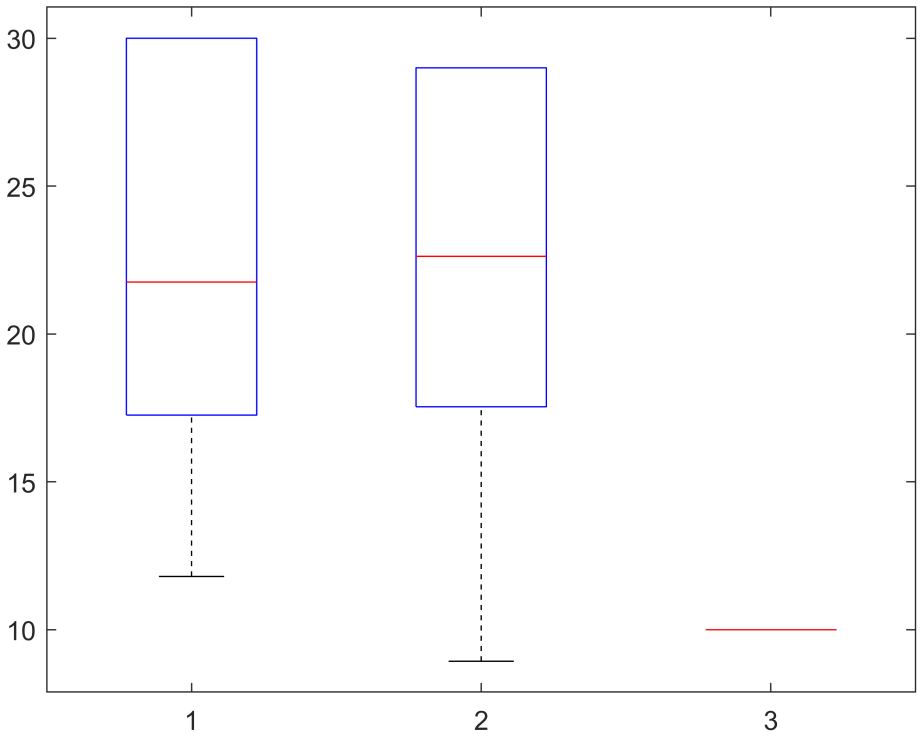
```
vecPayments = calcPayments(vecE,matL);
vecEquityNew = vecE + matPi'*vecPayments - vecPbar;

disp([{'Equity after clearing' 'Defaulted'};
    num2cell([vecEquityNew vecPayments < vecPbar])])
```

```
    'Equity after clearing'    'Defaulted'
    [               -5.4545]    [        1]
    [               -2.6364]    [        1]
    [                    30]    [        0]
```

We can now also simulate payoffs under stochastic paths for external assets, seeing that the first two agents repay their liabilities in full in some paths and default in the median path, the third agent never defaults in this model.

```
vecMu = [0.3;0.1;0.1];
matCovariance = [1 0 0;0 1 0;0 0 1];
numSimulations = 1000;
matValuations = calcValuations(vecE,vecMu,matCovariance,matL,numSimulations);
boxplot(matValuations')
```


30
25
20
15
10
1
2
3