In [1]:
```
import numpy as np
from scipy.optimize import minimize

A = np.array([[4, -5],
              [-2, 3]])

def fun(x, A):
    v  = x.T @ A
    return -np.max(v)

bounds = [(0, None), (0, None)]
constraints = [{'type': 'ineq', 'fun': lambda x: fun(x, A)},  # v >= 0
               {'type': 'eq', 'fun': lambda x: x[0] + x[1] - 1}]

# solve for player 1
results = minimize(fun, [0.5,0.5], args=(A), bounds=bounds, constraints=cons
print(f'player 1 probabilities: {results.x}')

# solve for player 2
A = A.T
results = minimize(fun, [0.5,0.5], args=(A), bounds=bounds, constraints=cons
print(f'player 2 probabilities: {results.x}')
```

```
player 1 probabilities: [0.35714286 0.64285714]
player 2 probabilities: [0.57142857 0.42857143]
```

In [2]:
```
!jupyter nbconvert --to html ./problem1.ipynb
```

```
[NbConvertApp] Converting notebook ./problem1.ipynb to html
[NbConvertApp] Writing 269321 bytes to problem1.html
```

In [ ]:
```
```