In [1]:
```
import math
import pandas as pd

results = [0, 0, 0]
R1 = [12, 12, 12, 8, 8, 8, 5, 5, 5]
R2 = [12, 8, 5, 12, 8, 5, 12, 8, 5]
AvgR = [12, 10, 8.5, 10, 8, 6.5, 8.5, 6.5, 5]
S = [0, 1, 1.75, 1, 2, 2.75, 1.75, 2.75, 3.5]
p = [0.57, 0.42, 0.25, 0.8, 0.4, 0.3, 0.92, 0.85, 0.72]
C1 = [6, 6, 6, 3, 3, 3, 2, 2, 2]
C2 = [8, 2, 1, 8, 2, 1, 8, 2, 1]

n = 3
Matrix = [[0] * n for i in range(n)]
k = 0
for i in range(3):
    for j in range(3):
        Matrix[i][j] = S[k] * (p[k] * (R1[k] - C1[k]) - (1 - p[k]) * (R2[k]
        k = k + 1
print(Matrix)
data = pd.DataFrame(Matrix, columns=['B1', 'B2', 'B3'], index=['A1', 'A2',

MinJ = [0, 0, 0]
MaxI = [0, 0, 0, 0]
for i in range(3):
    MinJ[i] = data.loc['A' + str(i + 1)].min()
for i in range(3):
    MaxI[i] = data['B' + str(i + 1)].max()
data['MinJ'] = MinJ
data.loc['MaxI'] = MaxI
print(data)
print("Нижняя граница игры = " + str(data['MinJ'].max()))
k = 1000
for i in data.loc['MaxI']:
    if k > i > 0:
        k = i
print("Верхняя граница игры = " + str(k))
print()
# 2-1
print("Задание 2")
print("Ситуация 1")
data1 = pd.DataFrame([[-2, 0, 3, -1, 1], [-1, 5, -2, -2, -1], [-3, -4, 0, -2
                      columns=['B1', 'B2', 'B3', 'B4', 'B5'], index=['A1', 'A

MinJ1 = [0, 0, 0, 0]
MaxI1 = [0, 0, 0, 0, 0, 0]
for i in range(4):
    MinJ1[i] = data1.loc['A' + str(i + 1)].min()
for i in range(5):
    MaxI1[i] = data1['B' + str(i + 1)].max()
data1['MinJ'] = MinJ1
data1.loc['MaxI'] = MaxI1
print(data1)
print("Нижняя граница игры = " + str(data1['MinJ'].max()))
k = 1000
for i in data1.loc['MaxI']:
```

```
    if k > i > 0:
        k = i
print("Верхняя граница игры = " + str(k))
if data1['MinJ'].max() == k:
    results[0] = k
print()
# 2-2
print('Ситуация 2')
data2 = pd.DataFrame([[4, -4, -1, 0], [7, 6, 2, 6], [5, 4, -6, 0]],
                     columns=['C1', 'C2', 'C3', 'C4'], index=['A1', 'A2', 'A
MinJ2 = [0, 0, 0]
MaxI2 = [0, 0, 0, 0, 0]
for i in range(3):
    MinJ2[i] = data2.loc['A' + str(i + 1)].min()
for i in range(4):
    MaxI2[i] = data2['C' + str(i + 1)].max()
data2['MinJ'] = MinJ2
data2.loc['MaxI'] = MaxI2
print(data2)
print("Нижняя граница игры = " + str(data2['MinJ'].max()))
k = 1000
for i in data2.loc['MaxI']:
    if k > i > 0:
        k = i
print("Верхняя граница игры = " + str(k))
if data1['MinJ'].max() == k:
    results[1] = k
print()

# 2-3
print('Ситауция 3')
data2 = pd.DataFrame([[-6, 5, -3, 2], [-3, 4, 3, -6], [-3, 7, 5, -3], [-3, -
                     columns=['D1', 'D2', 'D3', 'D4'], index=['A1', 'A2', 'A
MinJ1 = [0, 0, 0, 0, 0]
MaxI1 = [0, 0, 0, 0, 0]
for i in range(5):
    MinJ1[i] = data2.loc['A' + str(i + 1)].min()
for i in range(4):
    MaxI1[i] = data2['D' + str(i + 1)].max()
data2['MinJ'] = MinJ1
data2.loc['MaxI'] = MaxI1
print(data2)
g = -1000
for i in data2['MinJ']:
    if i > g and i != 0:
        g = i
print("Нижняя граница игры = " + str(g))
k = 1000
for i in data2.loc['MaxI']:
    if k > i != 0:
        k = i
print("Верхняя граница игры = " + str(k))
if g == k:
    results[2] = g

k = 0
for i in results:
```

```
    if i > k:
        k = i
print('Выйграшная ситуация ' + str(2))
```

```
[[0.0, -0.9600000000000004, -2.625], [3.2, -3.1999999999999993, -3.574999
9999999993], [4.2700000000000005, 4.537499999999999, 3.64]]
        B1      B2     B3   MinJ
A1    0.00 -0.9600 -2.625 -2.625
A2    3.20 -3.2000 -3.575 -3.575
A3    4.27  4.5375  3.640  3.640
MaxI  4.27  4.5375  3.640  0.000
Нижняя граница игры = 3.64
Верхняя граница игры = 3.64

Задание 2
Ситуация 1
      B1  B2  B3  B4  B5  MinJ
A1    -2   0   3  -1   1    -2
A2    -1   5  -2  -2  -1    -2
A3    -3  -4   0  -2  -2    -4
A4     3   5   3   3   1     1
MaxI   3   5   3   3   1     0
Нижняя граница игры = 1
Верхняя граница игры = 1

Ситуация 2
      C1  C2  C3  C4  MinJ
A1     4  -4  -1   0    -4
A2     7   6   2   6     2
A3     5   4  -6   0    -6
MaxI   7   6   2   6     0
Нижняя граница игры = 2
Верхняя граница игры = 2

Ситауция 3
      D1  D2  D3  D4  MinJ
A1    -6   5  -3   2    -6
A2    -3   4   3  -6    -6
A3    -3   7   5  -3    -3
A4    -3  -1  -4   8    -4
A5    -6   1  -6   5    -6
MaxI  -3   7   5   8     0
Нижняя граница игры = -3
Верхняя граница игры = -3
Выйграшная ситуация 2
```

In [ ]: