

## DOMINANT STRATEGIES (by Flavio Brienza)

/	Alfa	Beta
A	Aalfa ; alfaA	Abeta ; betaA
B	Balfa ; alfaB	Bbeta ; betaB

**Player 1** can choose either **A** or **B**.

**Player 2** can choose either **Alfa** or **Beta**.

The code is useful to find STRICTLY dominant strategies assuming that the higher the payoff, the better for the player.

```
pl1Aalfa = int(input('Enter A;alfa: '))
pl1Abeta = int(input('Enter A;beta: '))
pl1Balfa = int(input('Enter B;alfa: '))
pl1Bbeta = int(input('Enter B;beta: '))

pl2alfaA = int(input('Enter alfa;A: '))
pl2betaA = int(input('Enter beta;A: '))
pl2alfaB = int(input('Enter alfa;B: '))
pl2betaB = int(input('Enter beta;B: '))

def tryalfadominant() :
    if pl2alfaA > pl2betaA and pl2alfaB > pl2betaB:
        return 'alfa strictly dominates beta'

def trybetadominant():
    if (pl2betaA > pl2alfaA and pl2betaB > pl2alfaB):
        return 'beta strictly dominates alfa'

def tryAdominant():
    if pl1Aalfa > pl1Balfa and pl1Abeta > pl1Bbeta:
        return 'A strictly dominates B'

def tryBdominant():
    if pl1Balfa > pl1Aalfa and pl1Bbeta > pl1Abeta:
        return 'B strictly dominates A'

def solution1():
    if tryalfadominant() == 'alfa strictly dominates beta':
        return 'ALFA strictly dominates BETA'
    else:
        return ' '
```

```
def solution2():
    if trybetadominant() == 'beta strictly dominates alfa':
        return 'BETA strictly dominates ALFA'
    else:
        return ' '

def solution3():
    if tryAdominant() == 'A strictly dominates B':
        return 'A strictly dominates B'
    else:
        return ' '

def solution4():
    if tryBdominant() == 'B strictly dominates A':
        return 'B strictly dominates A'
    else:
        return ' '

def nodominantstrategies():
    if tryalfadominant() != 'alfa strictly dominates beta' and trybetadominant() != 'beta
strictly dominates alfa' and tryAdominant() != 'A strictly dominates B' and tryBdominant()
!= 'B strictly dominates A':
        return 'There are no dominant strategies'
    else:
        return ' '

print(solution1(), solution2(), solution3(), solution4(), nodominantstrategies())
```