Assignment 4

Michael Lee
CCST9017 - Hidden Order in Daily Life: A Mathematical Perspective
University Number 3035569110
Tutorial Group 009

September 27, 2018

Question 1a. Model the situation as a suitable cooperative game (N; v).

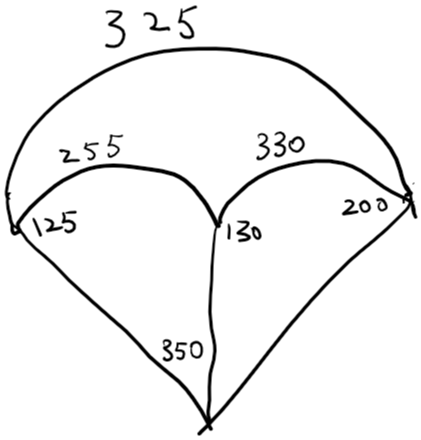
Answer 1a.

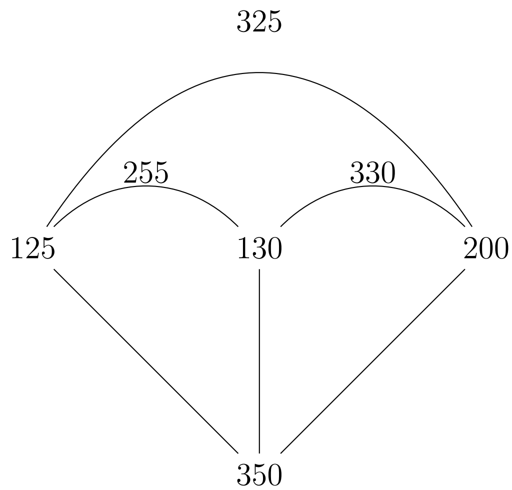
Let Amy, Betty and Clara be player 1, 2 and 3 respectively
Hence $N = \{1, 2, 3\}$
Such model should be

$$\begin{aligned}
v(\varnothing) &= 0 \\
v(1) &= 125 \\
v(2) &= 130 \\
v(3) &= 200 \\
v(1,2) &= 125 + 130 \\
&= 255 \\
v(2,3) &= 130 + 200 \\
&= 330 \\
v(1,3) &= 125 + 200 \\
&= 325 \\
v(1,2,3) &= 350
\end{aligned}$$

Question 1b. Use a diagram to represent the game defined in part a

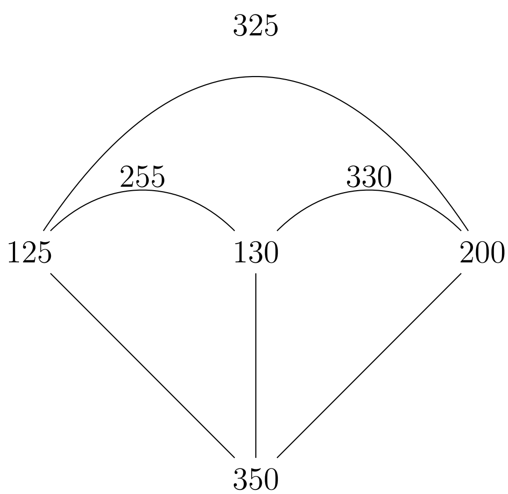
Answer 1b.



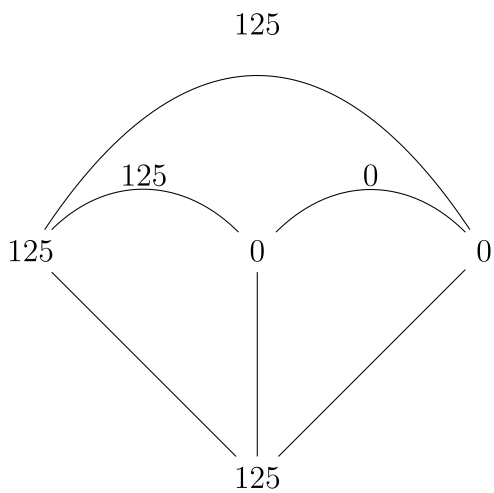


Question 1c. Find the Shapley value of the game by using two different methods.

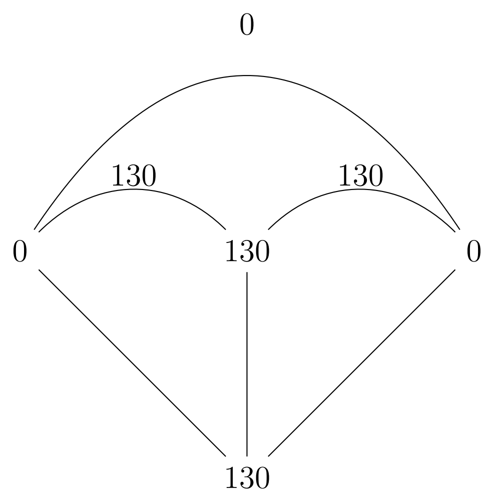
Answer 1c. First method: Spliting game



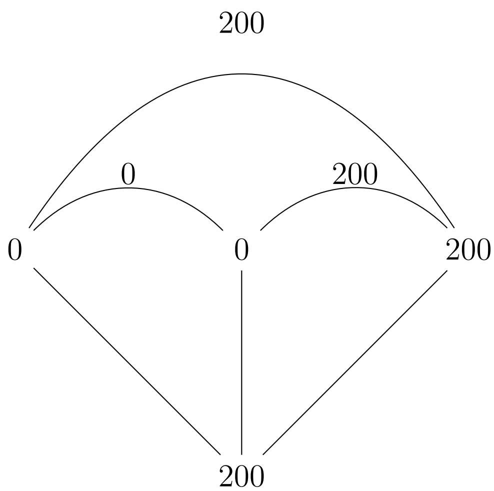
=



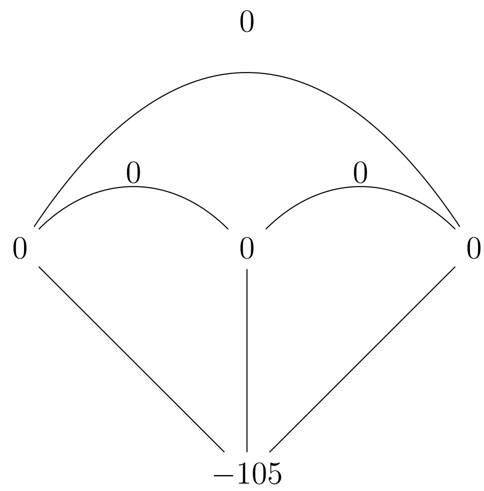
+



+



+



After the split, we can obtain the Shapley value of the game with

$$
\begin{array}{rrrr}
(& 125, & 0, & 0) \\
+(& 0, & 130, & 0) \\
+(& 0, & 0, & 200) \\
+(& -35, & -35, & -35) \\
=(& 90, & 95, & 165)
\end{array}
$$

Hence the Shapley value of this game is $(90, 95, 165)$

Answer 1c. Second method: Applying Shapley's Formula

To faciliate computation, I uses a Python script as shown below.

```python
from itertools import permutations
from math import factorial

# FronzenSet is used instead of Set
# because it can be used as key to dictionary
# Below defines certain results of v(S)
vDict = {
    frozenset(): 0,
    frozenset({1}): 125,
    frozenset({2}): 130,
    frozenset({3}): 200,
    frozenset({1, 2}): 255,
    frozenset({2, 3}): 330,
    frozenset({1, 3}): 325,
    frozenset({1, 2, 3}): 350
}

# Define the grand set G
G = {1, 2, 3}
n = len(G)
```

```python


# Define v(S) for that all S is a subset of G
def v(S):
    if S <= G:
        return vDict[S]
    else:
        raise ValueError('S is not subset of G')


# Define function N that for permutation p,
# return a frozenset that contains all elements in p before i
def N(p, i):
    return frozenset(p[:p.index(i)])


# Define function S to return the Shapley value of player i
def S(i):
    sum = 0
    for p in permutations(G):
        sum += v(N(p, i) | frozenset({i})) - v(N(p, i))
    return 1/factorial(n) * sum


# print the calculation result
print(S(1), S(2), S(3))
# 90.0 95.0 165.0
```

As the result of computation, it is verified that the Shapley value of this game is $(90, 95, 165)$ Therefore, with $90 + 95 + 165 = 350$, Amy, Betty and Clara should earn \$90, \$95 and \$165 respectively.

Question 2a. Model this situation as a game in coalitional form (N;v), where N=1,2,3 and v is the cost function.

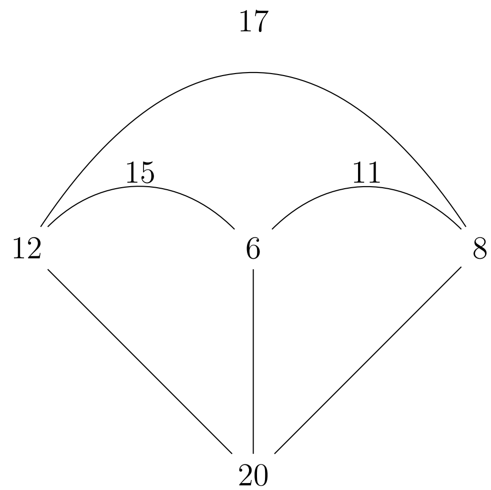
Answer 2a.

Let town A, B and C be player 1, 2 and 3 respectively,
cost function $v(S)$ is in a unit of million
Hence $N = \{1, 2, 3\}$
Such model should be

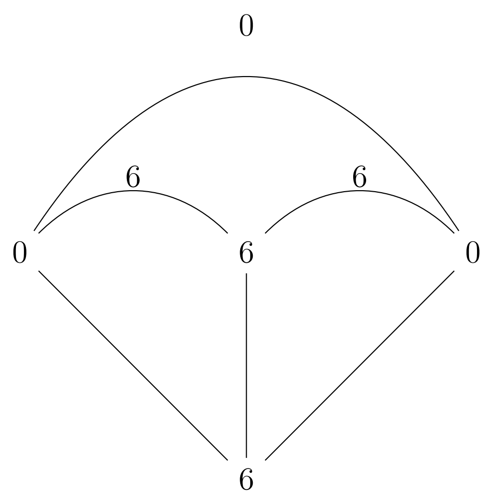
$$\begin{aligned}
v(\varnothing) &= 0 \\
v(1) &= 12 \\
v(2) &= 6 \\
v(3) &= 8 \\
v(1,2) &= 15 \\
v(2,3) &= 11 \\
v(1,3) &= 17 \\
v(1,2,3) &= 20
\end{aligned}$$

Question 2b. Assume the grand coalition N is formed. Find the Shapley value of the game by splitting the game into some suitable games.

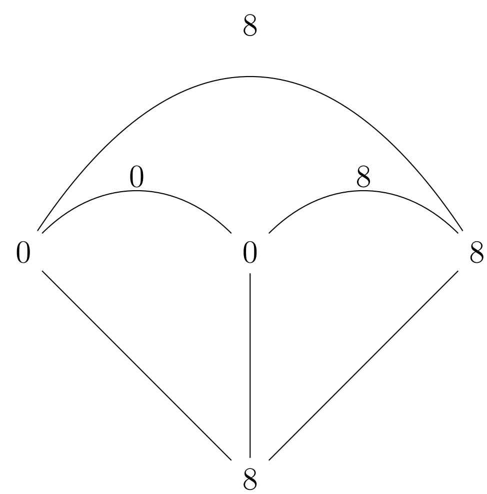
Answer 2b. Splitting the game below.



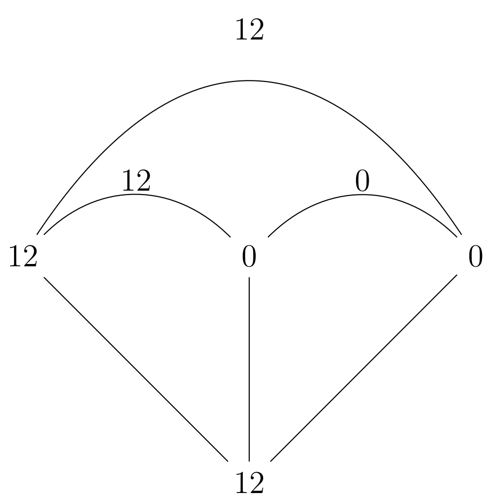
=



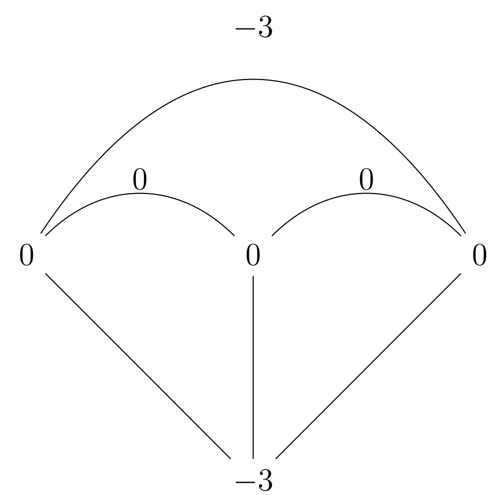
+

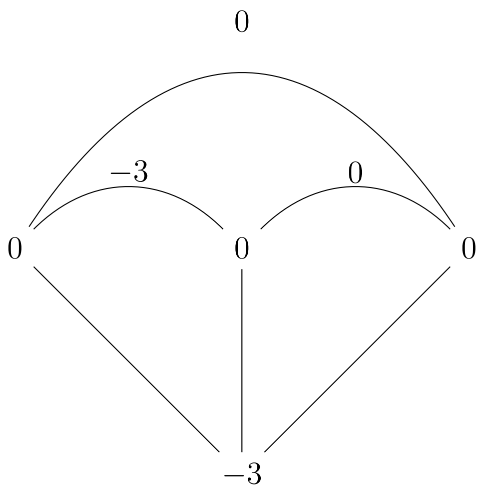


+

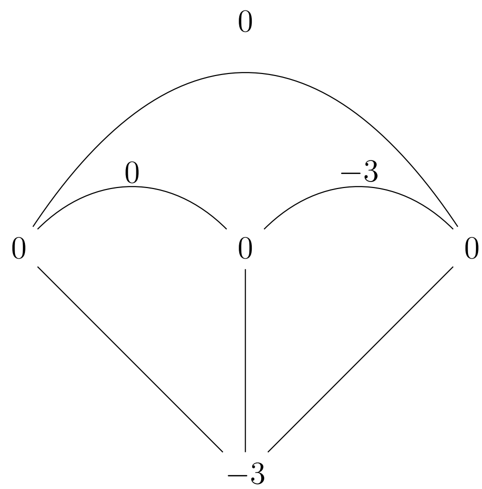


+

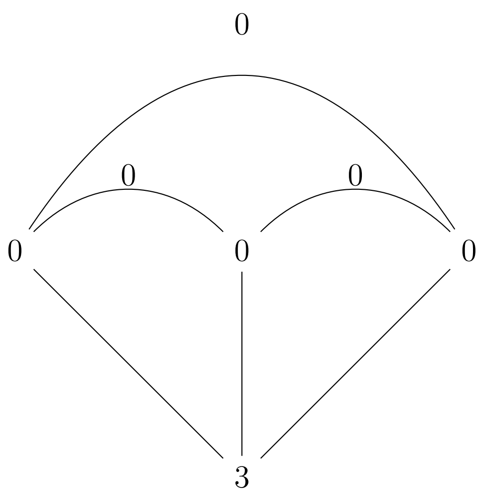




+



+



+

After the split, we can obtain the Shapley value of the game with

$$
\begin{array}{rrrr}
(& 0, & 6, & 0) \\
+(& 0, & 0, & 8) \\
+(& 12, & 0, & 0) \\
+(& -1.5, & 0, & -1.5) \\
+(& -1.5, & -1.5, & 0) \\
+(& 0, & -1.5, & -1.5) \\
+(& 1, & 1, & 1) \\
=(& 10, & 4, & 6)
\end{array}
$$

Hence the Shapley value of this game is $(10, 4, 6)$

Question 2c. Suppose A's cost is decreased from $12millionto11$ million and all the other costs are unchanged. Find the new Shapley value of the game.

Answer 2c.

The new model should be

$$
\begin{aligned}
v(\varnothing) &= 0 \\
v(1) &= 11 \\
v(2) &= 6 \\
v(3) &= 8 \\
v(1,2) &= 15 \\
v(2,3) &= 11 \\
v(1,3) &= 17 \\
v(1,2,3) &= 20
\end{aligned}
$$

To faciliate computation, I modified the Python script to as shown below.

```python
from itertools import permutations
from math import factorial

# FronzenSet is used instead of Set
# because it can be used as key to dictionary
# Below defines certain results of v(S)
vDict = {
    frozenset(): 0,
    frozenset({1}): 11,
    frozenset({2}): 6,
    frozenset({3}): 8,
    frozenset({1, 2}): 15,
    frozenset({2, 3}): 11,
    frozenset({1, 3}): 17,
    frozenset({1, 2, 3}): 20
}

# Define the grand set G
G = {1, 2, 3}
n = len(G)


# Define v(S) for that all S is a subset of G
def v(S):
    if S <= G:
        return vDict[S]
    else:
        raise ValueError('S is not subset of G')


# Define function N that for permutation p,
# return a frozenset that contains all elements in p before i
def N(p, i):
    return frozenset(p[:p.index(i)])


# Define function S to return the Shapley value of player i
def S(i):
```

```python
    sum = 0
    for p in permutations(G):
        sum += v(N(p, i) | frozenset({i})) - v(N(p, i))
    return 1/factorial(n) * sum


# print the calculation result
print(S(1), S(2), S(3))
# 9.666666666666666 4.166666666666666 6.166666666666666
```

The new Shapley value of this game is $(9\frac{2}{3}, 4\frac{1}{6}, 6\frac{1}{6})$