Graph Characteristics

Board Characteristics

- 6×6 Matrix
- Each square contains a number/color combination (eg. 1, Blue)
- Because it is a 6×6 matrix we chose numbers 1-6 and six colors, namely (blue, pink, white, green, red, & yellow)
- We use every possible permutation of the number/color combination to give us 6^2 squares.

Node Characteristics

- Tile Number
- Node Color
- Node Number

Edge Characteristics

- Edge exists iff
    - same node number
    - same node color
    - both nodes exist on same x or y axis

Let's get some basic imports out of the way

```
In [2]:  import matplotlib.pyplot as plt
         import networkx as nx
         import numpy as np
         import random
         import matplotlib.pyplot as plt
         import scipy as sp
```

Next, We structure our squares, tiles, and boards so we can visualize the boards.

Implementing classes with all necessary methods

In [3]:
```python
class Square:
    def __init__(self, color, number):
        self.color = color
        self.number = number

class Tile:
    def __init__(self, tile_number, matrix):
        self.number = tile_number
        # TODO: check if perfect square?
        self.matrix = matrix

    # since we always have a matrix and to make future operations simpler, we adjust
    # to return the integer which, when squared, forms the size of our matrix
    def get_shape(self):
        return self.matrix.shape[0]

    def print_tile(self):
        for row in self.matrix:
            for cell in row:
                print(cell.color, end=' ')
                print(cell.number, end=' ')
            print("\n")

    def print_tile_row(self, row):
        for cell in self.matrix[row]:
            print(cell.color, end=' ')
            print(cell.number, end=' | ')

    def get_tile_row(self, row):
        return self.matrix[row]

class Board:
    def __init__(self, matrix):
        # constructor will accept a matrix of Tiles.
        self.matrix = matrix
        self.graph = nx.Graph()
        self.adjacency_dict = {}
        # TODO: check if perfect square?

    def print_board(self):
        tile_shape = self.matrix[0][0].get_shape()

        for x in range(tile_shape):
            self.matrix[0][0].print_tile_row(x)
            self.matrix[0][1].print_tile_row(x)
            print("\n")

        for x in range(tile_shape):
            self.matrix[1][0].print_tile_row(x)
            self.matrix[1][1].print_tile_row(x)
            print("\n")

    # deconstructs the board into a matrix of squares
    def deconstruct_board(self):
        tile_shape = self.matrix[0][0].get_shape()
```

```python
            deconstructed_matrix = []
            for x in range(tile_shape):
                row = np.concatenate([self.matrix[0][0].get_tile_row(x), self.matr
ix[0][1].get_tile_row(x)])
                deconstructed_matrix.append(row)

            for x in range(tile_shape):
                row = np.concatenate([self.matrix[1][0].get_tile_row(x), self.matr
ix[1][1].get_tile_row(x)])
                deconstructed_matrix.append(row)

            return deconstructed_matrix

    def build_graph(self):
        deconstructed_matrix = np.array(self.deconstruct_board()).reshape(6,6)
        # make each square a graph node
        for row in deconstructed_matrix:
            for elem in row:
                label = elem.color + str(elem.number)
                self.graph.add_node(elem, label=label)

        # for each node draw edges
        for row_index, row in enumerate(deconstructed_matrix, start=0):
            for col_index, elem in enumerate(row, start=0):
                # examine column, if same number or color, create an edge
 between the two nodes
                # library handles case of duplicate edges
                for square in deconstructed_matrix[:,col_index]:
                    if(square.color == elem.color or square.number == elem
.number):
                        self.graph.add_edge(square,elem)

                # examine row, if same number or color, create an edge bet
ween the two nodes
                # library handles case of duplicate edges
                for square in deconstructed_matrix[row_index]:
                    if(square.color == elem.color or square.number == elem
.number):
                        self.graph.add_edge(square,elem)

        self.graph.remove_edges_from(nx.selfloop_edges(self.graph))

    def print_graph(self):
        labels = nx.get_node_attributes(self.graph, 'label')
        pos = nx.spring_layout(self.graph,k=0.75,iterations=20)
        plt.figure(3,figsize=(10,6))
        nx.draw(self.graph, pos, labels=labels, node_size=500, font_weight='bo
ld', node_color='y')
        plt.show()

    def get_laplacian_matrix(self):
        return nx.laplacian_matrix(self.graph)

    def get_adjacency_matrix(self):
        return nx.adjacency_matrix(self.graph)

    def print_graph_characteristics(self):
```

```python
        print("Graph characteristics: ")
        print(self.graph.number_of_nodes(), "nodes")
        print(self.graph.number_of_edges(), "edges")
        print("Connected?", nx.is_connected(self.graph))

    def set_adjacency_dict(self):
        if self.graph.number_of_nodes() == 0:
            return "ERROR"

        adjacency_dict = {}

        for n, nbrdict in self.graph.adjacency():
            adjacency_dict[n.color + str(n.number)] = []
            for item in nbrdict:
                adjacency_dict[n.color + str(n.number)].append(item.color + st
r(item.number))

        self.adjacency_dict = adjacency_dict

    def print_adjacency_dict(self):
        if self.graph.number_of_nodes() == 0:
            return "ERROR"

        adjacency_dict = {}

        for n, nbrdict in self.graph.adjacency():
            adjacency_dict[n.color + str(n.number)] = []
            for item in nbrdict:
                adjacency_dict[n.color + str(n.number)].append(item.color + st
r(item.number))

        #print adjancecy dict for verfication of correctness
        for k, v in adjacency_dict.items():
            print (k, '-->', v)
```

We have the key classes implemented. Let's build the actual microrobots board.

```
In [4]:  def build_actual_blackside_microrobots_board():
             tileOne = Tile(1, np.array([[Square("W", 6), Square("B", 6), Square("B", 2
         )],
                     [Square("R", 3), Square("Y", 4),Square("W", 3)],
                     [Square("B", 3), Square("G", 4), Square("W", 5)]], dtype=np.objec
         t).reshape(3,3))

             tileTwo = Tile(2, np.array([[Square("P", 5), Square("P", 2), Square("P", 3
         )],
                         [Square("R", 6), Square("R", 4),Square("Y", 3)],
                         [Square("B", 4), Square("G", 5), Square("G", 3)]], dtype=np.o
         bject).reshape(3,3))

             tileThree = Tile(3, np.array([[Square("B", 1), Square("R", 2), Square("W",
         4)],
                         [Square("W", 1), Square("G", 2),Square("G", 6)],
                         [Square("Y", 2), Square("W", 2), Square("P", 4)]], dtype=np.o
         bject).reshape(3,3))

             tileFour = Tile(4, np.array([[Square("P", 1), Square("Y", 6), Square("Y",
         1)],
                         [Square("Y", 5), Square("B", 5),Square("R", 5)],
                         [Square("P", 6), Square("R", 1), Square("G", 1)]], dtype=np.o
         bject).reshape(3,3))

             return Board(np.array([[tileOne, tileTwo], [tileThree, tileFour]], dtype=n
         p.object).reshape(2,2))

         blackside_board = build_actual_blackside_microrobots_board()

         blackside_board.print_board()
```

```
W 6 | B 6 | B 2 | P 5 | P 2 | P 3 |

R 3 | Y 4 | W 3 | R 6 | R 4 | Y 3 |

B 3 | G 4 | W 5 | B 4 | G 5 | G 3 |

B 1 | R 2 | W 4 | P 1 | Y 6 | Y 1 |

W 1 | G 2 | G 6 | Y 5 | B 5 | R 5 |

Y 2 | W 2 | P 4 | P 6 | R 1 | G 1 |
```

Now, we implement the ability to generate random boards

In [5]:
```python
# takes n numbers and n colors and generates a random n x n microrobots board
def generate_random_board(numbers, colors):
        # TODO check if lists are equal in size, error if not
        # TODO generalize to n x n by defining tile size rather than just usin
g 3x3

        # build hashmap to store numbers and colors
        expected_matrix_size = len(numbers) * len(colors)
        hashmap = {}
        for color in colors:
            hashmap[color] = []
            for num in numbers:
                hashmap[color].append(num)

        hashmap_copy = hashmap.copy()

        # generates a random unique (exactly once) set of combinations of
        # numbers and colors and returns them as an instance of Square
        squares = []
        colors_copy = colors.copy()
        while len(colors_copy) > 0:
            # chose random color
            rand_color = random.choice(colors_copy)
            rand_num = random.choice(hashmap_copy[rand_color])
            hashmap_copy[rand_color].pop(hashmap_copy[rand_color].index(rand_n
um))

            squares.append(Square(rand_color, rand_num))

            # now we check if we used last num, if so remove color
            # care, this is terminating condition on while loop
            if len(hashmap_copy[rand_color]) == 0:
                colors_copy.pop(colors_copy.index(rand_color))


        # doouble check if len is as expected
        if len(squares) != expected_matrix_size:
            print("ERROR: You only produced ", len(squares), "squares and we e
xpected", expected_matrix_size)
            return

        # generate 4 arrays of 9
        # TODO: the generation of these tiles is obvious not, itself, random.
 need to decide if this matters
        split_squares = np.array_split(squares, 4)
        board = Board(np.array([[Tile(1, split_squares[0].reshape(3,3)), Tile(
2,split_squares[1].reshape(3,3))],
                    [Tile(3,split_squares[2].reshape(3,3)), Tile(4, split_squares
[3].reshape(3,3))]], dtype=np.object).reshape(2,2))

        return board

random_board = generate_random_board([1,2,3,4,5,6], ["W", "R", "B", "P", "Y",
"G"])
random_board.print_board()
```

```
B 6 | Y 2 | P 4 | B 4 | B 3 | W 3 |

P 2 | Y 5 | P 3 | W 5 | G 5 | G 3 |

R 5 | W 1 | G 2 | G 6 | Y 1 | Y 4 |

Y 6 | G 4 | G 1 | R 3 | W 2 | R 6 |

Y 3 | B 1 | W 6 | B 2 | R 1 | B 5 |

P 5 | P 1 | W 4 | R 2 | R 4 | P 6 |
```

Now, let's explore the connectedness of a board.

We consider two squares connected iff

```
- They exist on the same row OR column AND
- They are the the same color OR number
```

Here, we draw the resulting graph of the ACTUAL blackside microrobots board and label it with the associated node values
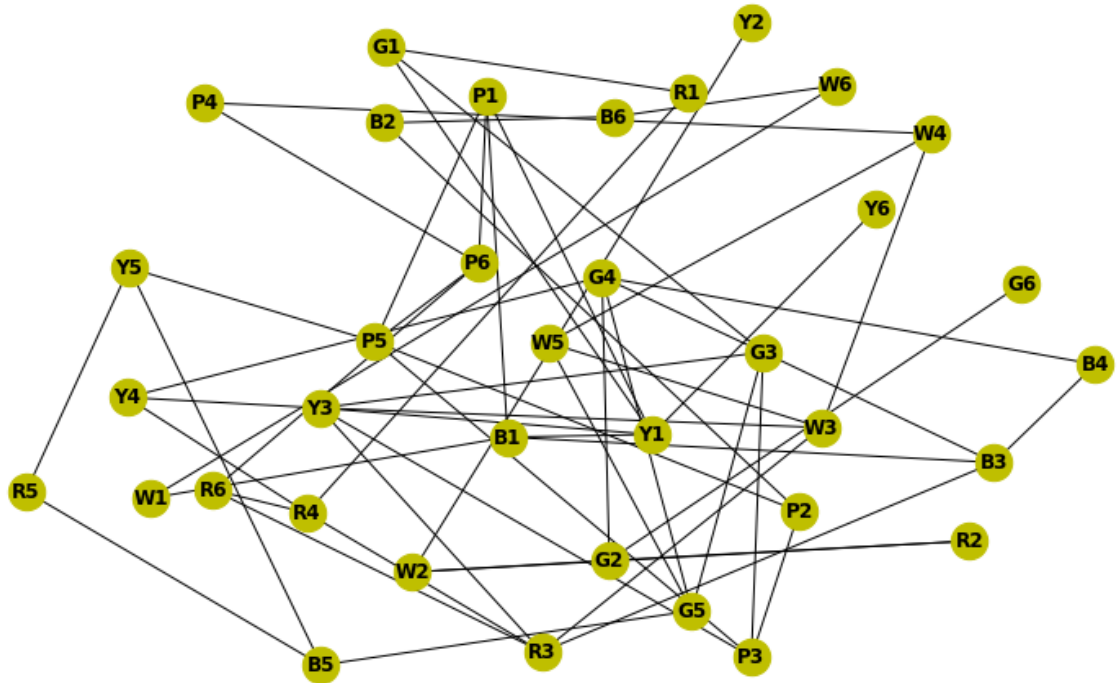
```
In [6]:  blackside_board.build_graph()
         blackside_board.print_graph_characteristics()
         blackside_board.print_graph()
```

Graph characteristics:
36 nodes
58 edges
Connected? True

Just to make sure we got everything right let's take a look at the adjacency list.

In [7]:
```
blackside_board.set_adjacency_dict()
blackside_board.print_adjacency_dict()
```

```
W6 --> ['W1', 'B6']
B6 --> ['W6', 'B2']
B2 --> ['B6', 'P2']
P5 --> ['P1', 'Y5', 'P6', 'P2', 'P3']
P2 --> ['B2', 'P5', 'P3']
P3 --> ['P5', 'P2', 'Y3', 'G3']
R3 --> ['B3', 'W3', 'R6', 'R4', 'Y3']
Y4 --> ['G4', 'R4', 'Y3']
W3 --> ['R3', 'W5', 'W4', 'Y3']
R6 --> ['R3', 'P6', 'R4']
R4 --> ['R3', 'Y4', 'R6', 'R1']
Y3 --> ['P3', 'R3', 'Y4', 'W3', 'G3', 'Y1']
B3 --> ['R3', 'B1', 'B4', 'G3']
G4 --> ['Y4', 'G2', 'B4', 'G5', 'G3']
W5 --> ['W3', 'W4', 'G5']
B4 --> ['B3', 'G4']
G5 --> ['G4', 'W5', 'B5', 'G3']
G3 --> ['P3', 'Y3', 'B3', 'G4', 'G5', 'G1']
B1 --> ['B3', 'W1', 'P1', 'Y1']
R2 --> ['G2', 'W2']
W4 --> ['W3', 'W5', 'P4']
P1 --> ['P5', 'B1', 'P6', 'Y1']
Y6 --> ['Y1']
Y1 --> ['Y3', 'B1', 'P1', 'Y6', 'G1']
W1 --> ['W6', 'B1']
G2 --> ['G4', 'R2', 'W2', 'G6']
G6 --> ['G2']
Y5 --> ['P5', 'B5', 'R5']
B5 --> ['G5', 'Y5', 'R5']
R5 --> ['Y5', 'B5']
Y2 --> ['W2']
W2 --> ['R2', 'G2', 'Y2']
P4 --> ['W4', 'P6']
P6 --> ['P5', 'R6', 'P1', 'P4']
R1 --> ['R4', 'G1']
G1 --> ['G3', 'Y1', 'R1']
```

Alright, we compared that to the actual board and nothing looks sus!

TESTING RANDOMLY GENERATED THEORY

- We questioned whether, just by nature, these boards tend to be connected even when randomly generated. The answer to that question is no.

In [ ]:

```python
%%time

# generates n random boards with the given numbers & colors arrays
def generate_random_boards(n, numbers, colors):
    boards_list = []
    for x in range(n):
        boards_list.append(generate_random_board(numbers, colors))
    return boards_list

nums = [1,2,3,4,5,6]
cols = ["W", "R", "B", "P", "Y", "G"]
boards_list = generate_random_boards(100, nums, cols)

count_not_connected = 0

for b in boards_list:
    print("\n")
    print(" -------------------------------- ")
    print("\n")
    b.print_board()
    b.build_graph()
    b.print_graph_characteristics()

    if not nx.is_connected(b.graph):
        b.print_graph()
        b.set_adjacency_dict()
        b.print_adjacency_dict()
        count_not_connected = count_not_connected + 1

print("    ==========================================    ")

print(count_not_connected, " out of ", len(boards_list)," graphs were not conn
ected")
```

P 3 | B 3 | P 2 | R 1 | B 5 | P 1 |

B 6 | B 2 | Y 3 | W 3 | G 5 | Y 6 |

R 5 | R 3 | W 2 | B 1 | Y 2 | P 6 |

P 5 | W 6 | R 6 | Y 4 | Y 5 | W 5 |

P 4 | W 4 | G 1 | R 4 | W 1 | G 6 |

G 4 | B 4 | G 3 | Y 1 | G 2 | R 2 |

Graph characteristics:
36 nodes
59 edges
Connected? True

G 6 | B 5 | B 3 | R 5 | Y 1 | R 3 |

R 2 | Y 4 | B 1 | W 2 | G 2 | G 3 |

G 4 | W 6 | P 3 | B 4 | P 2 | Y 2 |

W 3 | R 1 | W 5 | G 5 | B 6 | B 2 |

W 4 | Y 6 | Y 3 | P 6 | R 6 | P 1 |

Y 5 | W 1 | G 1 | P 5 | P 4 | R 4 |

Graph characteristics:
36 nodes
45 edges
Connected? False
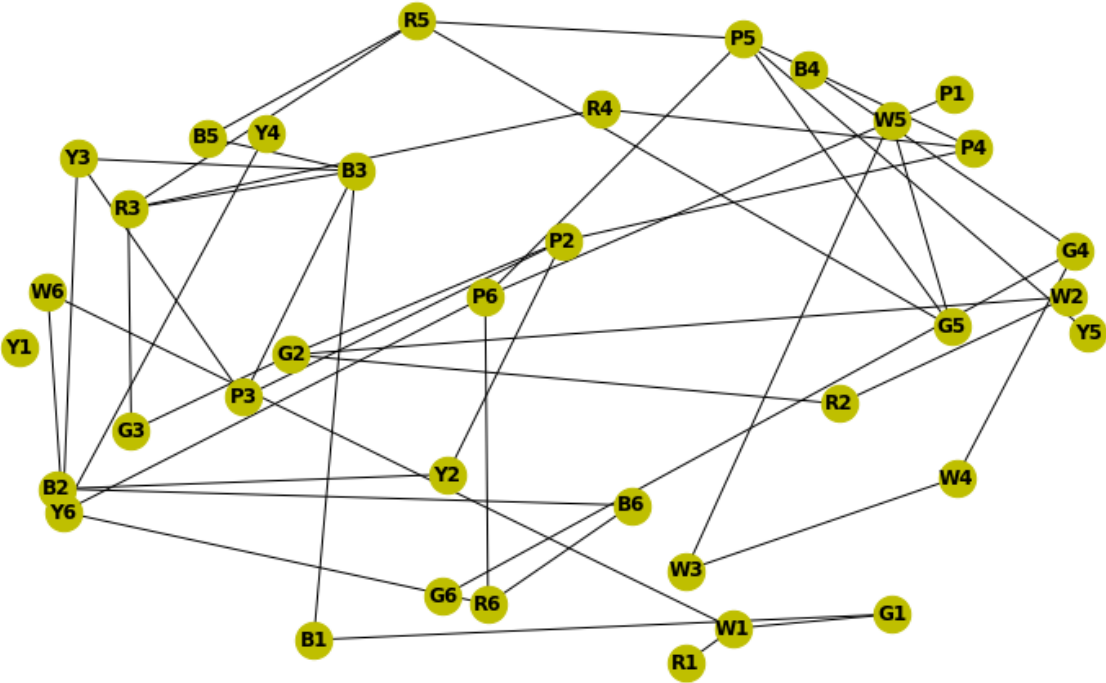
```
G6 --> ['G4']
B5 --> ['B3', 'R5']
B3 --> ['B5', 'B1', 'P3', 'Y3', 'R3']
R5 --> ['B5', 'G5', 'P5', 'R3']
Y1 --> []
R3 --> ['B3', 'R5', 'G3', 'R4']
R2 --> ['W2', 'G2']
Y4 --> ['Y6']
B1 --> ['B3', 'G1']
W2 --> ['R2', 'G2']
G2 --> ['R2', 'W2', 'P2', 'G3']
G3 --> ['R3', 'G2']
G4 --> ['G6', 'W4', 'B4']
W6 --> ['Y6', 'W1']
P3 --> ['B3', 'Y3', 'P2']
B4 --> ['G4']
P2 --> ['G2', 'P3', 'P4', 'Y2']
Y2 --> ['P2', 'B2']
W3 --> ['W4', 'W5']
R1 --> ['W1']
W5 --> ['W3', 'G5']
G5 --> ['R5', 'W5', 'P5']
B6 --> ['R6', 'B2']
B2 --> ['Y2', 'B6']
W4 --> ['G4', 'W3']
Y6 --> ['Y4', 'W6', 'Y3', 'P6', 'R6']
Y3 --> ['B3', 'P3', 'Y6']
P6 --> ['Y6', 'P5', 'R6', 'P1']
R6 --> ['B6', 'Y6', 'P6']
P1 --> ['P6']
Y5 --> ['P5']
W1 --> ['W6', 'R1', 'G1']
G1 --> ['B1', 'W1']
P5 --> ['R5', 'G5', 'P6', 'Y5', 'P4']
P4 --> ['P2', 'P5', 'R4']
R4 --> ['R3', 'P4']
```

```
        ---------------------------------


Y 2 | P 3 | B 4 | Y 4 | B 6 | W 3 |

R 3 | W 1 | Y 1 | B 3 | B 1 | Y 3 |

B 5 | P 6 | G 6 | B 2 | Y 5 | G 3 |

Y 6 | P 2 | W 2 | G 1 | G 4 | W 5 |

G 5 | R 4 | R 1 | W 4 | P 4 | P 1 |

G 2 | R 6 | P 5 | R 5 | W 6 | R 2 |

Graph characteristics:
36 nodes
50 edges
Connected? True
```
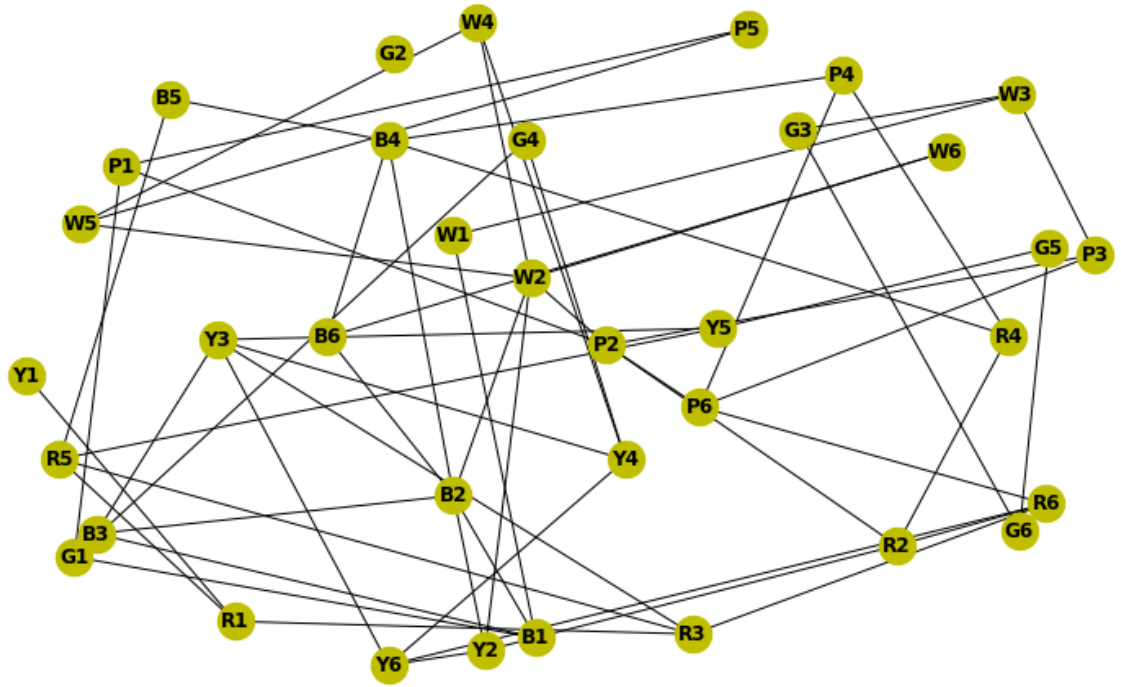
```
         ----------------------------------

         Y 1 | B 1 | B 2 | W 6 | W 1 | Y 3 |

         B 3 | R 6 | W 3 | B 6 | B 5 | W 2 |

         R 2 | R 1 | G 4 | W 4 | Y 6 | B 4 |

         G 1 | G 5 | G 6 | Y 5 | P 3 | Y 4 |

         G 2 | R 5 | G 3 | P 5 | Y 2 | P 4 |

         R 4 | P 1 | W 5 | P 6 | R 3 | P 2 |
```

Graph characteristics:
36 nodes
59 edges
Connected? True


```
         ----------------------------------

         Y 1 | G 2 | B 4 | B 5 | P 4 | R 4 |

         G 3 | Y 6 | Y 2 | W 3 | R 6 | R 2 |

         R 1 | Y 3 | W 6 | R 5 | R 3 | Y 5 |

         G 6 | B 3 | B 2 | W 1 | B 1 | G 5 |

         P 5 | Y 4 | B 6 | G 4 | G 1 | P 1 |

         W 5 | W 4 | W 2 | P 3 | P 6 | P 2 |
```

Graph characteristics:
36 nodes
58 edges
Connected? False

```
Y1 --> ['R1']
G2 --> []
B4 --> ['B2', 'B6', 'B5', 'P4', 'R4']
B5 --> ['B4', 'R5']
P4 --> ['B4', 'P6', 'R4']
R4 --> ['B4', 'P4', 'R2']
G3 --> ['G6', 'W3']
Y6 --> ['Y3', 'Y4', 'Y2', 'R6']
Y2 --> ['Y6', 'B2', 'W2', 'R2']
W3 --> ['G3', 'W1', 'P3']
R6 --> ['Y6', 'R3', 'P6', 'R2']
R2 --> ['R4', 'Y2', 'R6', 'P2']
R1 --> ['Y1', 'R5', 'R3']
Y3 --> ['Y6', 'B3', 'Y4', 'R3', 'Y5']
W6 --> ['B6', 'W2']
R5 --> ['B5', 'R1', 'R3', 'Y5']
R3 --> ['R6', 'R1', 'Y3', 'R5']
Y5 --> ['Y3', 'R5', 'G5']
G6 --> ['G3', 'G5']
B3 --> ['Y3', 'B2', 'B1']
B2 --> ['B4', 'Y2', 'B3', 'B6', 'W2', 'B1']
W1 --> ['W3', 'B1']
B1 --> ['B3', 'B2', 'W1', 'G1']
G5 --> ['Y5', 'G6']
P5 --> ['W5', 'P1']
Y4 --> ['Y6', 'Y3', 'W4', 'G4']
B6 --> ['B4', 'W6', 'B2']
G4 --> ['Y4', 'G1']
G1 --> ['B1', 'G4', 'P1']
P1 --> ['P5', 'G1', 'P2']
W5 --> ['P5', 'W4', 'W2']
W4 --> ['Y4', 'W5', 'W2']
W2 --> ['Y2', 'W6', 'B2', 'W5', 'W4', 'P2']
P3 --> ['W3', 'P6', 'P2']
P6 --> ['P4', 'R6', 'P3', 'P2']
P2 --> ['R2', 'P1', 'W2', 'P3', 'P6']
```

```
W 3 | W 4 | P 3 | W 6 | W 5 | B 3 |

W 1 | R 4 | G 3 | R 3 | Y 3 | G 2 |

R 2 | W 2 | B 5 | B 6 | G 6 | Y 4 |

G 5 | G 4 | Y 6 | R 6 | G 1 | P 4 |

P 5 | R 1 | P 6 | B 2 | B 4 | Y 2 |

R 5 | P 1 | Y 1 | P 2 | B 1 | Y 5 |
```
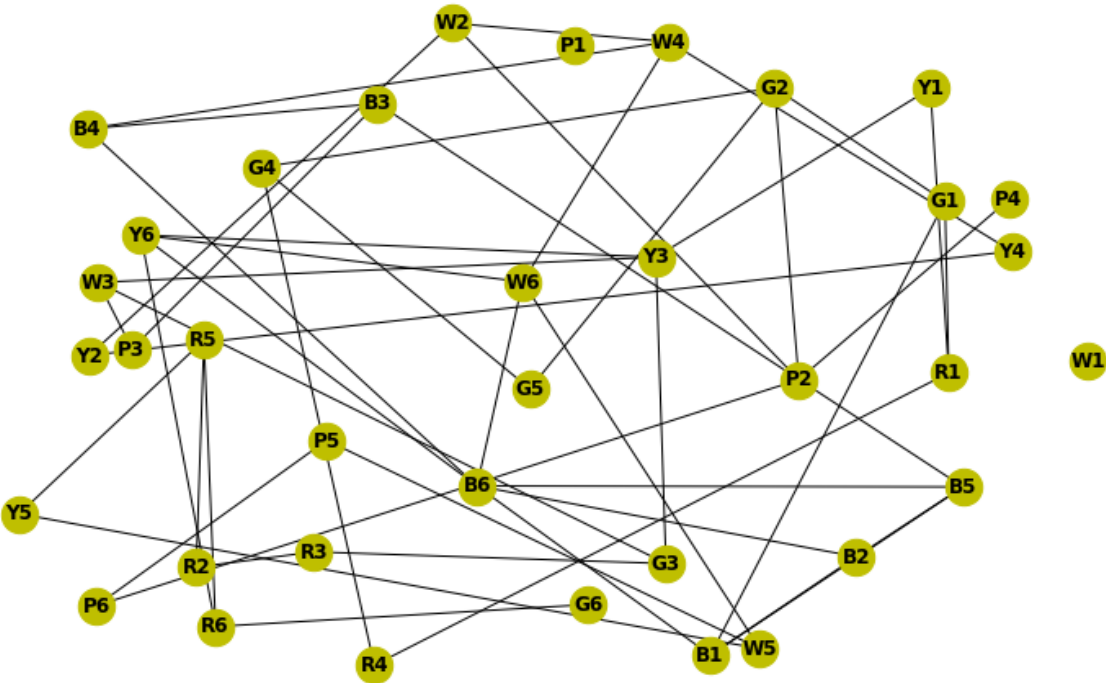
Graph characteristics:
36 nodes
60 edges
Connected? True

```
----------------------------------
```

P 4 | B 6 | B 2 | B 5 | B 1 | R 3 |

Y 1 | B 4 | R 5 | B 3 | P 3 | R 2 |

W 2 | W 4 | G 6 | Y 2 | Y 4 | P 5 |

P 2 | G 5 | W 1 | G 4 | G 2 | P 6 |

R 1 | W 6 | Y 5 | R 4 | G 1 | W 5 |

Y 3 | Y 6 | R 6 | P 1 | W 3 | G 3 |

Graph characteristics:
36 nodes
49 edges
Connected? False

```
           P4 --> ['P2']
           B6 --> ['B4', 'W6', 'Y6', 'B2', 'B5', 'B1']
           B2 --> ['B6', 'B5', 'B1']
           B5 --> ['B6', 'B2', 'B3', 'B1']
           B1 --> ['B6', 'B2', 'B5', 'G1']
           R3 --> ['R2', 'G3']
           Y1 --> ['R1', 'Y3']
           B4 --> ['B6', 'W4', 'B3']
           R5 --> ['Y5', 'R6', 'R2']
           B3 --> ['B5', 'B4', 'P3']
           P3 --> ['B3', 'W3']
           R2 --> ['R3', 'R5']
           W2 --> ['P2', 'W4', 'Y2']
           W4 --> ['B4', 'W2', 'W6', 'Y4']
           G6 --> ['R6']
           Y2 --> ['W2', 'Y4']
           Y4 --> ['W4', 'Y2']
           P5 --> ['P6', 'W5']
           P2 --> ['P4', 'W2', 'G2', 'P6']
           G5 --> ['G4', 'G2']
           W1 --> []
           G4 --> ['G5', 'R4', 'G2']
           G2 --> ['P2', 'G5', 'G4', 'G1']
           P6 --> ['P5', 'P2']
           R1 --> ['Y1', 'R4', 'G1']
           W6 --> ['B6', 'W4', 'Y6', 'W5']
           Y5 --> ['R5', 'W5']
           R4 --> ['G4', 'R1']
           G1 --> ['B1', 'G2', 'R1']
           W5 --> ['P5', 'W6', 'Y5']
           Y3 --> ['Y1', 'Y6', 'W3', 'G3']
           Y6 --> ['B6', 'W6', 'Y3', 'R6']
           R6 --> ['R5', 'G6', 'Y6']
           P1 --> []
           W3 --> ['P3', 'Y3', 'G3']
           G3 --> ['R3', 'Y3', 'W3']


           ---------------------------------


           W 5 | B 5 | W 1 | G 3 | P 5 | G 2 |

           R 4 | B 2 | B 6 | G 5 | G 1 | R 1 |

           Y 1 | W 2 | Y 4 | G 4 | P 1 | R 3 |

           G 6 | W 3 | Y 6 | W 6 | B 4 | Y 5 |

           P 2 | W 4 | R 2 | P 3 | P 4 | Y 2 |

           Y 3 | R 5 | R 6 | B 1 | B 3 | P 6 |

           Graph characteristics:
           36 nodes
           53 edges
           Connected? True
```

R 1 | P 5 | B 6 | G 2 | R 5 | R 4 |

G 4 | Y 5 | B 5 | P 6 | Y 6 | P 3 |

B 1 | W 3 | P 1 | B 4 | B 2 | W 2 |

G 6 | R 3 | Y 2 | G 5 | P 2 | Y 4 |

G 3 | B 3 | W 6 | P 4 | Y 3 | R 6 |

Y 1 | G 1 | W 1 | W 5 | W 4 | R 2 |

Graph characteristics:
36 nodes
52 edges
Connected? True

R 5 | G 1 | Y 3 | Y 5 | G 4 | P 6 |

P 1 | P 2 | Y 6 | P 4 | R 4 | Y 2 |

R 6 | B 3 | G 3 | R 2 | G 5 | G 6 |

R 3 | Y 1 | R 1 | B 1 | W 1 | W 3 |

G 2 | P 5 | Y 4 | B 2 | B 6 | W 5 |

P 3 | W 6 | W 2 | W 4 | B 4 | B 5 |

Graph characteristics:
36 nodes
54 edges
Connected? True

R 1 | R 6 | P 1 | G 5 | B 3 | G 3 |

Y 1 | W 2 | W 3 | G 6 | P 6 | G 4 |

R 5 | R 3 | R 2 | W 5 | G 2 | Y 2 |

Y 3 | Y 5 | B 4 | B 6 | W 1 | W 6 |
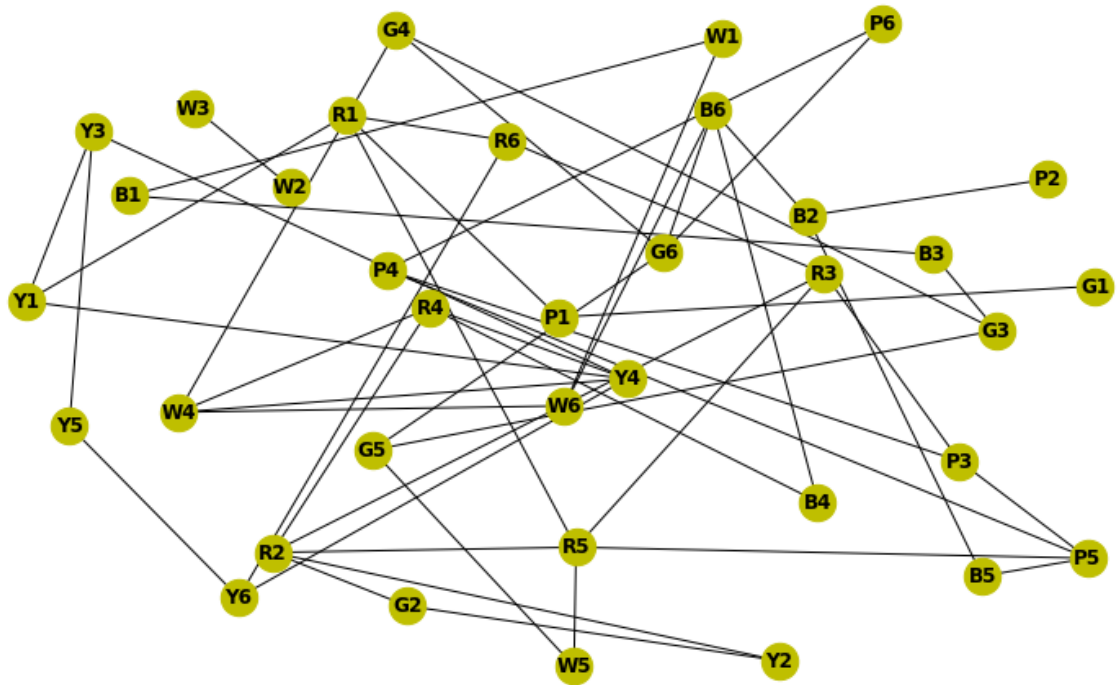
Y 4 | Y 6 | R 4 | P 2 | B 1 | W 4 |

P 5 | P 3 | G 1 | B 2 | P 4 | B 5 |

Graph characteristics:
36 nodes
51 edges
Connected? False

```
R1 --> ['Y1', 'R5', 'R6', 'P1']
R6 --> ['R1', 'R3', 'Y6']
P1 --> ['R1', 'G1']
G5 --> ['G6', 'W5', 'G3']
B3 --> ['B1', 'G3']
G3 --> ['G5', 'B3', 'G4']
Y1 --> ['R1', 'Y3', 'Y4']
W2 --> ['W3']
W3 --> ['W2']
G6 --> ['G5', 'B6', 'P6', 'G4']
P6 --> ['G6', 'P4']
G4 --> ['G3', 'G6', 'W4']
R5 --> ['R1', 'P5', 'R3', 'R2', 'W5']
R3 --> ['R6', 'R5', 'P3', 'R2']
R2 --> ['R5', 'R3', 'R4', 'G2', 'Y2']
W5 --> ['G5', 'R5']
G2 --> ['R2', 'Y2']
Y2 --> ['R2', 'G2']
Y3 --> ['Y1', 'Y4', 'Y5']
Y5 --> ['Y3', 'Y6']
B4 --> ['R4', 'B6']
B6 --> ['G6', 'B4', 'B2', 'W6']
W1 --> ['B1', 'W6']
W6 --> ['B6', 'W1', 'W4']
Y4 --> ['Y1', 'Y3', 'Y6', 'R4', 'W4']
Y6 --> ['R6', 'Y5', 'Y4']
R4 --> ['R2', 'B4', 'Y4', 'W4']
P2 --> ['B2']
B1 --> ['B3', 'W1']
W4 --> ['G4', 'W6', 'Y4', 'R4']
P5 --> ['R5', 'P3', 'P4', 'B5']
P3 --> ['R3', 'P5', 'P4']
G1 --> ['P1']
B2 --> ['B6', 'P2', 'B5']
P4 --> ['P6', 'P5', 'P3']
B5 --> ['P5', 'B2']
```

```
          ---------------------------------
```

```
G 6 | W 2 | W 3 | B 6 | W 6 | R 1 |

Y 3 | W 5 | Y 5 | Y 6 | Y 2 | W 1 |

B 1 | W 4 | R 2 | G 5 | B 4 | R 3 |

Y 1 | B 2 | B 3 | G 1 | G 4 | P 1 |

R 5 | P 5 | R 4 | B 5 | G 2 | P 6 |

Y 4 | R 6 | G 3 | P 3 | P 4 | P 2 |
```
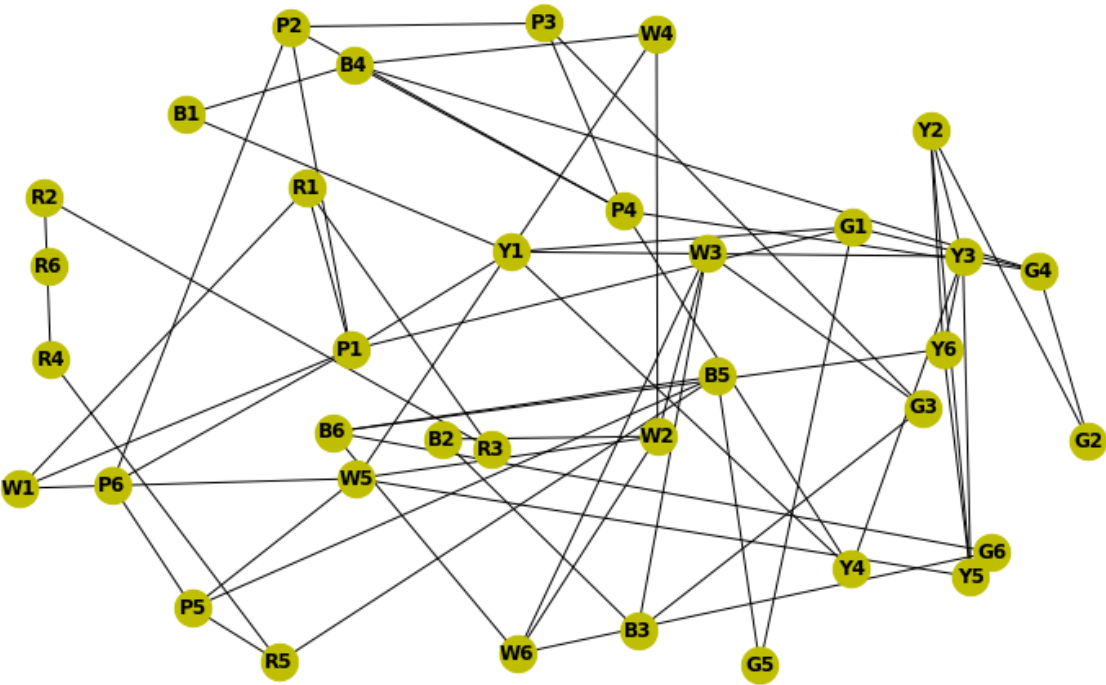
```
Graph characteristics:
36 nodes
61 edges
Connected? False
```

```
        G6 --> ['B6', 'W6']
        W2 --> ['W5', 'W4', 'B2', 'W3', 'W6']
        W3 --> ['W2', 'B3', 'G3', 'W6']
        B6 --> ['G6', 'Y6', 'B5', 'W6']
        W6 --> ['G6', 'W2', 'W3', 'B6']
        R1 --> ['W1', 'R3', 'P1']
        Y3 --> ['Y1', 'Y4', 'Y5', 'Y6', 'Y2']
        W5 --> ['W2', 'W4', 'P5', 'Y5', 'W1']
        Y5 --> ['Y3', 'W5', 'Y6', 'Y2']
        Y6 --> ['B6', 'Y3', 'Y5', 'Y2']
        Y2 --> ['Y3', 'Y5', 'Y6', 'G2']
        W1 --> ['R1', 'W5', 'P1']
        B1 --> ['Y1', 'B4']
        W4 --> ['W2', 'W5', 'B4']
        R2 --> ['R4', 'R3']
        G5 --> ['G1', 'B5']
        B4 --> ['B1', 'W4', 'G4', 'P4']
        R3 --> ['R1', 'R2']
        Y1 --> ['Y3', 'B1', 'Y4', 'G1', 'P1']
        B2 --> ['W2', 'B3']
        B3 --> ['W3', 'B2', 'G3']
        G1 --> ['G5', 'Y1', 'G4', 'P1']
        G4 --> ['B4', 'G1', 'G2', 'P4']
        P1 --> ['R1', 'W1', 'Y1', 'G1', 'P6', 'P2']
        R5 --> ['P5', 'R4', 'B5']
        P5 --> ['W5', 'R5', 'B5', 'P6']
        R4 --> ['R2', 'R5']
        B5 --> ['B6', 'G5', 'R5', 'P5']
        G2 --> ['Y2', 'G4']
        P6 --> ['P1', 'P5', 'P2']
        Y4 --> ['Y3', 'Y1', 'P4']
        R6 --> []
        G3 --> ['W3', 'B3', 'P3']
        P3 --> ['G3', 'P4', 'P2']
        P4 --> ['B4', 'G4', 'Y4', 'P3', 'P2']
        P2 --> ['P1', 'P6', 'P3', 'P4']


          ---------------------------------


        W 3 | R 3 | B 2 | R 6 | W 2 | Y 6 |

        W 6 | W 5 | P 6 | Y 4 | Y 2 | R 1 |

        R 4 | P 2 | Y 1 | R 5 | G 2 | R 2 |

        Y 5 | W 1 | G 3 | P 3 | P 5 | B 3 |

        W 4 | G 4 | G 6 | B 4 | G 1 | P 4 |

        B 6 | B 5 | P 1 | Y 3 | G 5 | B 1 |

        Graph characteristics:
        36 nodes
        58 edges
        Connected? True
```

```
-----------------------------------

R 1 | G 5 | W 6 | P 1 | P 5 | G 3 |

B 3 | W 4 | P 3 | R 4 | W 2 | R 6 |

R 3 | R 2 | G 2 | Y 1 | P 6 | B 6 |

R 5 | W 3 | B 2 | B 1 | Y 6 | W 5 |

G 1 | W 1 | G 6 | B 5 | B 4 | P 4 |

P 2 | Y 3 | G 4 | Y 2 | Y 5 | Y 4 |
```

Graph characteristics:
36 nodes
51 edges
Connected? True


```
-----------------------------------

R 1 | P 1 | Y 1 | R 5 | Y 6 | Y 2 |

Y 5 | G 5 | G 4 | Y 4 | G 2 | P 5 |

P 4 | R 3 | G 1 | G 3 | Y 3 | G 6 |

W 2 | R 6 | W 1 | P 6 | B 4 | P 3 |

R 2 | W 4 | R 4 | B 3 | W 3 | W 5 |

B 5 | P 2 | B 2 | B 6 | B 1 | W 6 |
```

Graph characteristics:
36 nodes
58 edges
Connected? False
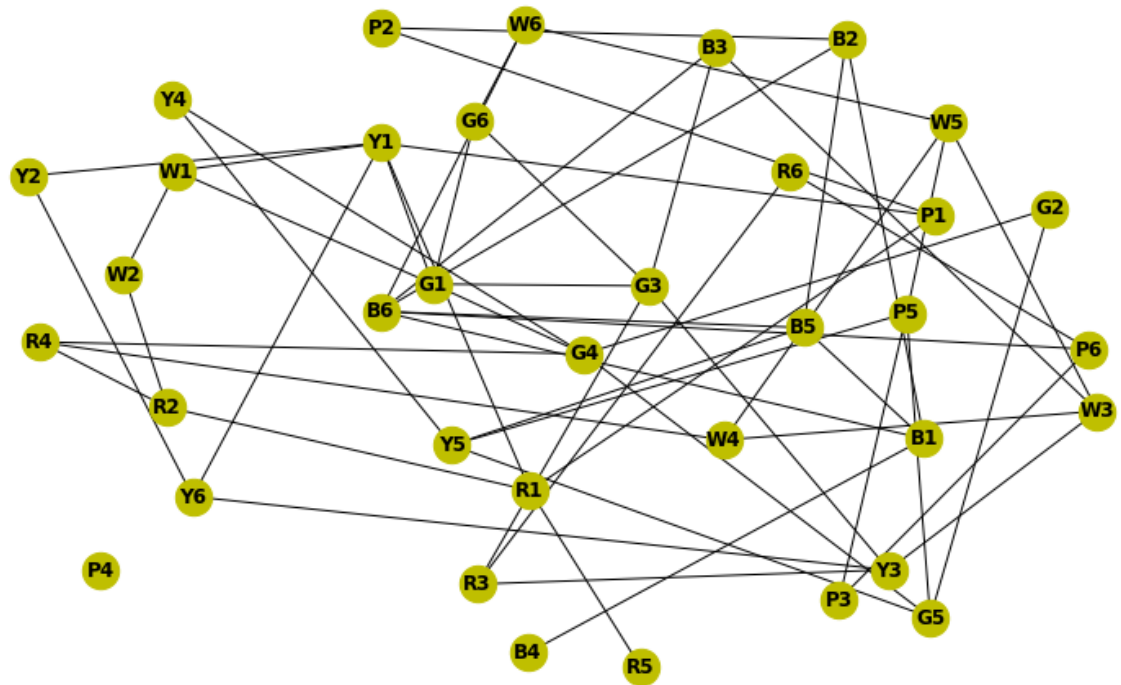
```
        R1 --> ['R2', 'P1', 'Y1', 'R5']
        P1 --> ['R1', 'P2', 'Y1']
        Y1 --> ['R1', 'P1', 'G1', 'W1', 'Y6', 'Y2']
        R5 --> ['R1']
        Y6 --> ['Y1', 'Y3', 'Y2']
        Y2 --> ['Y1', 'Y6']
        Y5 --> ['B5', 'G5', 'Y4', 'P5']
        G5 --> ['Y5', 'G4', 'G2', 'P5']
        G4 --> ['G5', 'G1', 'R4', 'Y4', 'G2']
        Y4 --> ['Y5', 'G4']
        G2 --> ['G5', 'G4']
        P5 --> ['Y5', 'G5', 'P3', 'W5']
        P4 --> []
        R3 --> ['R6', 'G3', 'Y3']
        G1 --> ['Y1', 'G4', 'W1', 'G3', 'G6']
        G3 --> ['R3', 'G1', 'B3', 'Y3', 'G6']
        Y3 --> ['Y6', 'R3', 'G3', 'W3']
        G6 --> ['G1', 'G3', 'W6']
        W2 --> ['R2', 'W1']
        R6 --> ['R3', 'P6']
        W1 --> ['Y1', 'G1', 'W2']
        P6 --> ['R6', 'B6', 'P3']
        B4 --> ['B1']
        P3 --> ['P5', 'P6']
        R2 --> ['R1', 'W2', 'R4']
        W4 --> ['R4', 'W3', 'W5']
        R4 --> ['G4', 'R2', 'W4']
        B3 --> ['G3', 'B6', 'W3']
        W3 --> ['Y3', 'W4', 'B3', 'W5']
        W5 --> ['P5', 'W4', 'W3', 'W6']
        B5 --> ['Y5', 'B2', 'B6', 'B1']
        P2 --> ['P1', 'B2']
        B2 --> ['B5', 'P2', 'B6', 'B1']
        B6 --> ['P6', 'B3', 'B5', 'B2', 'B1', 'W6']
        B1 --> ['B4', 'B5', 'B2', 'B6']
        W6 --> ['G6', 'W5', 'B6']


        ---------------------------------


        Y 2 | R 5 | B 2 | G 1 | Y 3 | P 2 |

        G 3 | B 5 | Y 5 | R 6 | W 1 | G 4 |

        P 1 | W 5 | G 6 | G 2 | W 6 | P 4 |

        G 5 | Y 1 | Y 4 | B 6 | B 1 | R 2 |

        B 4 | Y 6 | W 3 | P 3 | R 1 | R 3 |

        P 6 | P 5 | B 3 | W 4 | W 2 | R 4 |

        Graph characteristics:
        36 nodes
        48 edges
        Connected? False
```
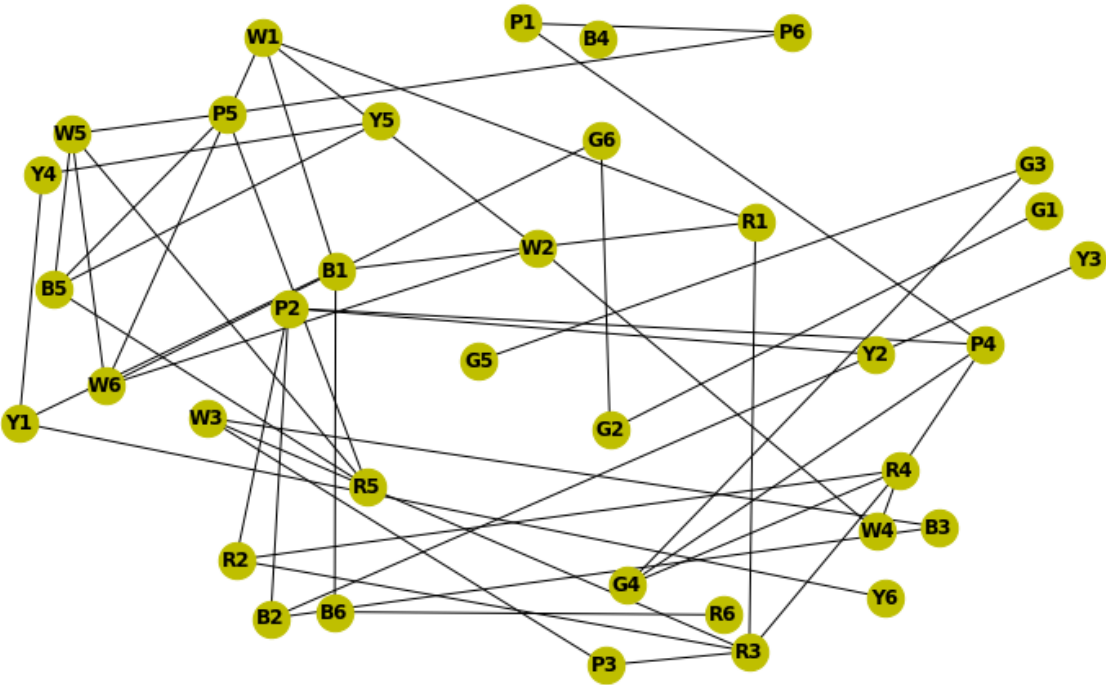
```
        Y2 --> ['B2', 'Y3', 'P2']
        R5 --> ['B5', 'W5', 'P5']
        B2 --> ['Y2', 'B3', 'P2']
        G1 --> ['G2']
        Y3 --> ['Y2']
        P2 --> ['Y2', 'B2', 'P4', 'R2']
        G3 --> ['G5', 'G4']
        B5 --> ['R5', 'W5', 'P5', 'Y5']
        Y5 --> ['B5', 'Y4']
        R6 --> ['B6']
        W1 --> ['W6', 'B1', 'R1', 'W2']
        G4 --> ['G3', 'P4', 'R4']
        P1 --> ['P6', 'P4']
        W5 --> ['R5', 'B5', 'P5', 'W6']
        G6 --> ['G2', 'W6']
        G2 --> ['G1', 'G6']
        W6 --> ['W1', 'W5', 'G6', 'W2']
        P4 --> ['P2', 'G4', 'P1', 'R4']
        G5 --> ['G3']
        Y1 --> ['Y6', 'Y4', 'B1']
        Y4 --> ['Y5', 'Y1']
        B6 --> ['R6', 'B1']
        B1 --> ['W1', 'Y1', 'B6', 'R1']
        R2 --> ['P2', 'R3', 'R4']
        B4 --> []
        Y6 --> ['Y1']
        W3 --> ['B3', 'P3', 'R3']
        P3 --> ['W3', 'R3']
        R1 --> ['W1', 'B1', 'R3']
        R3 --> ['R2', 'W3', 'P3', 'R1', 'R4']
        P6 --> ['P1', 'P5']
        P5 --> ['R5', 'B5', 'W5', 'P6']
        B3 --> ['B2', 'W3']
        W4 --> ['W2', 'R4']
        W2 --> ['W1', 'W6', 'W4']
        R4 --> ['G4', 'P4', 'R2', 'R3', 'W4']


        ---------------------------------


    Y 2 | R 1 | B 3 | G 5 | P 1 | B 5 |

    B 1 | G 1 | Y 5 | Y 3 | B 6 | W 3 |

    W 2 | Y 1 | R 2 | G 6 | Y 4 | R 6 |

    B 4 | G 2 | G 3 | W 5 | Y 6 | R 4 |

    B 2 | G 4 | W 4 | R 3 | P 6 | W 6 |

    R 5 | W 1 | P 3 | P 2 | P 5 | P 4 |

    Graph characteristics:
    36 nodes
    55 edges
    Connected? True
```

```
            ----------------------------------

            G 6 | Y 3 | R 6 | Y 6 | B 3 | W 2 |

            R 1 | Y 4 | B 2 | R 3 | B 4 | B 1 |

            P 1 | P 2 | P 4 | P 3 | P 5 | G 2 |

            B 6 | W 5 | B 5 | W 4 | R 2 | R 5 |

            R 4 | G 3 | G 4 | Y 2 | W 3 | Y 1 |

            P 6 | G 1 | G 5 | W 1 | W 6 | Y 5 |
```

Graph characteristics:
36 nodes
60 edges
Connected? True

```
            ----------------------------------

            W 3 | Y 1 | P 2 | G 5 | P 5 | R 5 |

            Y 2 | P 4 | B 4 | G 1 | Y 4 | B 2 |

            Y 3 | R 4 | Y 5 | B 6 | B 5 | P 6 |

            G 3 | R 1 | Y 6 | R 3 | W 4 | W 1 |

            B 1 | P 1 | W 6 | W 2 | R 2 | G 4 |

            W 5 | B 3 | P 3 | R 6 | G 2 | G 6 |
```

Graph characteristics:
36 nodes
46 edges
Connected? True

```
            ----------------------------------

            Y 2 | B 4 | B 6 | R 2 | P 6 | P 2 |

            Y 4 | Y 6 | R 4 | B 2 | Y 5 | R 5 |

            G 1 | P 1 | W 1 | W 2 | Y 3 | P 3 |

            B 3 | W 4 | R 3 | Y 1 | B 5 | B 1 |

            R 6 | W 5 | W 6 | W 3 | G 4 | G 3 |
```

R 1 | P 5 | P 4 | G 6 | G 5 | G 2 |

Graph characteristics:
36 nodes
58 edges
Connected? True

G 3 | B 5 | R 2 | Y 3 | W 1 | B 4 |

G 5 | R 3 | G 1 | R 6 | R 4 | Y 1 |

B 3 | P 1 | B 2 | P 5 | B 1 | P 4 |

P 3 | W 2 | Y 2 | W 3 | W 4 | G 6 |

P 6 | G 4 | Y 4 | W 5 | W 6 | Y 5 |

B 6 | P 2 | R 5 | G 2 | Y 6 | R 1 |

Graph characteristics:
36 nodes
53 edges
Connected? False
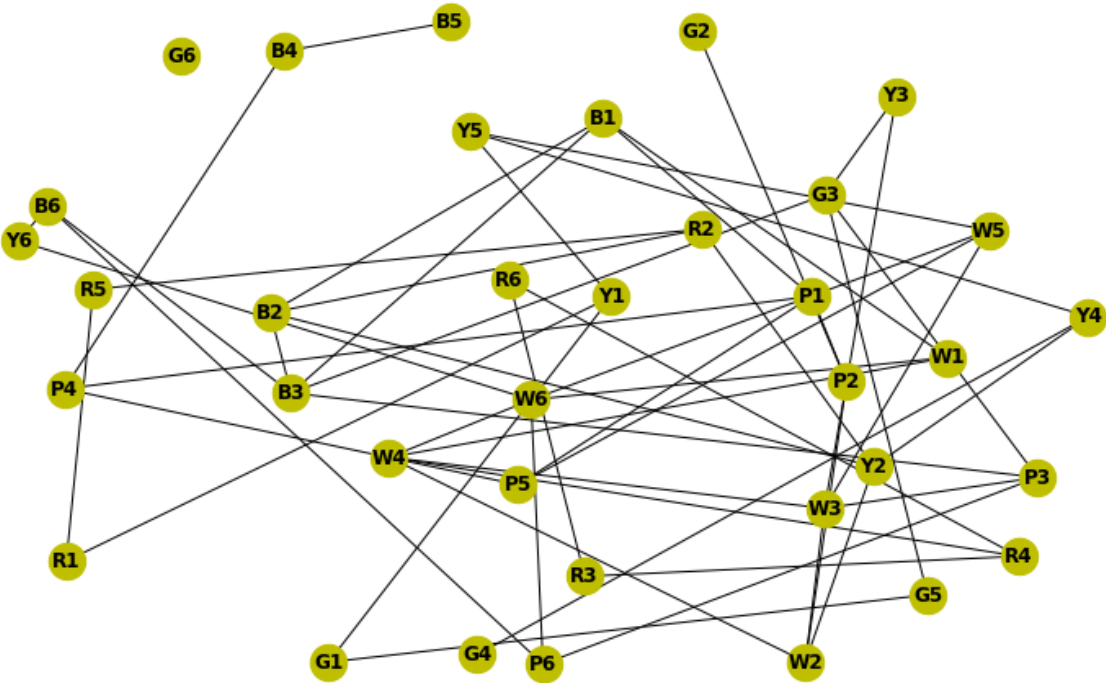
```
G3 --> ['G5', 'B3', 'P3', 'Y3']
B5 --> ['B4']
R2 --> ['B2', 'Y2', 'R5']
Y3 --> ['G3', 'W3']
W1 --> ['B1', 'W4', 'W6']
B4 --> ['B5', 'P4']
G5 --> ['G3', 'G1']
R3 --> ['R6', 'R4']
G1 --> ['G5', 'Y1']
R6 --> ['R3', 'R4']
R4 --> ['R3', 'R6', 'W4']
Y1 --> ['G1', 'Y5', 'R1']
B3 --> ['G3', 'P3', 'B6', 'B2', 'B1']
P1 --> ['P2', 'P5', 'B1', 'P4']
B2 --> ['R2', 'B3', 'Y2', 'B1']
P5 --> ['P1', 'W5', 'P4']
B1 --> ['W1', 'B3', 'P1', 'B2']
P4 --> ['B4', 'P1', 'P5']
P3 --> ['G3', 'B3', 'P6', 'W3']
W2 --> ['P2', 'Y2', 'W3', 'W4']
Y2 --> ['R2', 'B2', 'W2', 'Y4']
W3 --> ['Y3', 'P3', 'W2', 'W5', 'W4']
W4 --> ['W1', 'R4', 'W2', 'W3', 'W6']
G6 --> []
P6 --> ['P3', 'B6', 'W6']
G4 --> ['Y4']
Y4 --> ['Y2', 'G4', 'Y5']
W5 --> ['P5', 'W3', 'W6', 'Y5']
W6 --> ['W1', 'W4', 'P6', 'W5', 'Y6']
Y5 --> ['Y1', 'Y4', 'W5']
B6 --> ['B3', 'P6', 'Y6']
P2 --> ['P1', 'W2', 'G2']
R5 --> ['R2', 'R1']
G2 --> ['P2']
Y6 --> ['W6', 'B6']
R1 --> ['Y1', 'R5']
```

```
          ---------------------------------

     Y 4 | B 6 | R 3 | P 1 | B 5 | B 3 |

     G 6 | B 1 | P 6 | G 3 | B 2 | G 5 |

     B 4 | R 1 | P 4 | Y 6 | P 2 | G 1 |

     G 4 | P 5 | W 4 | W 3 | Y 3 | Y 5 |

     W 5 | R 2 | G 2 | W 1 | W 6 | R 4 |

     Y 2 | P 3 | Y 1 | W 2 | R 6 | R 5 |
```

```
Graph characteristics:
36 nodes
49 edges
Connected? False
```
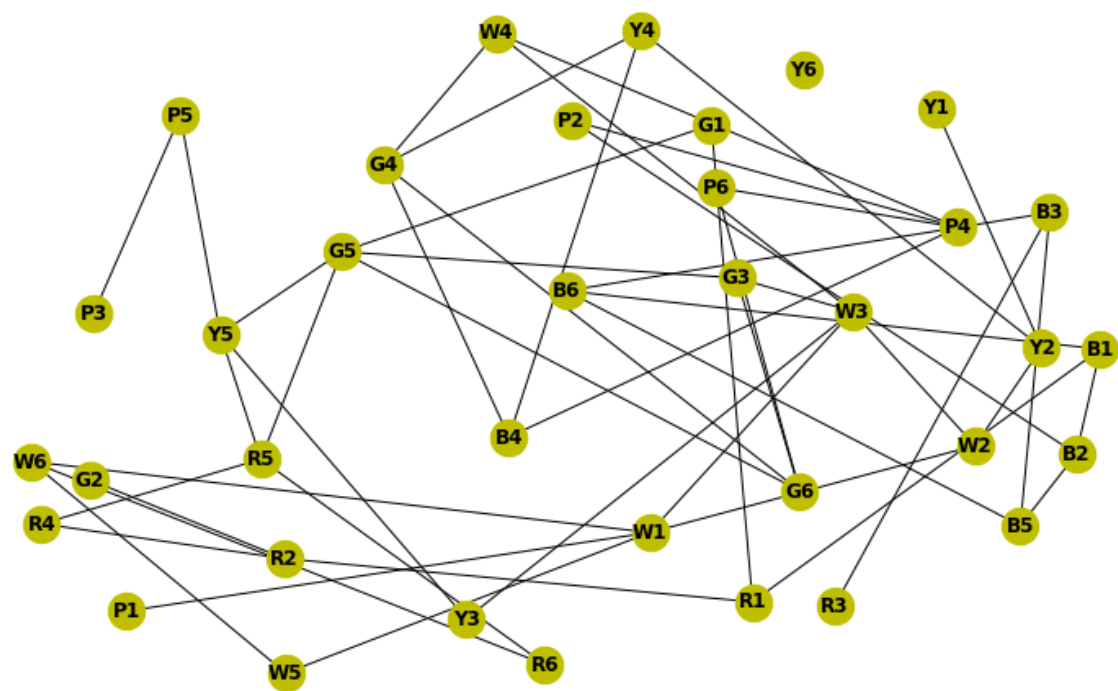
```
Y4 --> ['B4', 'G4', 'Y2']
B6 --> ['B1', 'B5', 'B3']
R3 --> ['B3']
P1 --> ['W1']
B5 --> ['B6', 'B2', 'B3']
B3 --> ['B6', 'R3', 'B5']
G6 --> ['G4', 'P6', 'G3', 'G5']
B1 --> ['B6', 'R1', 'B2']
P6 --> ['G6', 'P4']
G3 --> ['G6', 'W3', 'G5']
B2 --> ['B5', 'B1', 'P2']
G5 --> ['G6', 'G3', 'G1', 'Y5', 'R5']
B4 --> ['Y4', 'G4', 'P4']
R1 --> ['B1', 'R2', 'G1']
P4 --> ['P6', 'B4', 'W4', 'P2']
Y6 --> []
P2 --> ['B2', 'P4']
G1 --> ['G5', 'R1']
G4 --> ['Y4', 'G6', 'B4', 'W4']
P5 --> ['P3', 'Y5']
W4 --> ['P4', 'G4', 'W3']
W3 --> ['G3', 'W4', 'W1', 'W2', 'Y3']
Y3 --> ['W3', 'Y5']
Y5 --> ['G5', 'P5', 'Y3', 'R5']
W5 --> ['W1', 'W6']
R2 --> ['R1', 'G2', 'R4']
G2 --> ['R2']
W1 --> ['P1', 'W3', 'W5', 'W2', 'W6']
W6 --> ['W5', 'W1', 'R6']
R4 --> ['R2', 'R5']
Y2 --> ['Y4', 'Y1', 'W2']
P3 --> ['P5']
Y1 --> ['Y2']
W2 --> ['W3', 'W1', 'Y2']
R6 --> ['W6', 'R5']
R5 --> ['G5', 'Y5', 'R4', 'R6']
```

```
        ---------------------------------

    G 5 | W 5 | W 4 | G 6 | Y 1 | G 3 |

    P 5 | W 1 | B 1 | P 2 | Y 6 | G 4 |

    Y 5 | R 1 | P 6 | Y 3 | Y 2 | Y 4 |

    G 2 | P 1 | W 6 | B 5 | R 2 | B 6 |

    P 3 | R 4 | P 4 | W 2 | R 3 | R 6 |

    R 5 | G 1 | W 3 | B 3 | B 4 | B 2 |
```

Graph characteristics:
36 nodes
60 edges
Connected? True

```
-----------------------------------

P 4 | B 6 | P 2 | B 2 | W 1 | P 6 |

R 3 | W 6 | W 4 | W 3 | Y 4 | G 6 |

Y 5 | R 4 | B 4 | R 2 | W 5 | G 5 |

G 3 | B 5 | R 1 | B 1 | P 1 | Y 6 |

G 1 | P 3 | G 4 | B 3 | P 5 | R 5 |

R 6 | Y 1 | W 2 | G 2 | Y 3 | Y 2 |
```

Graph characteristics:
36 nodes
59 edges
Connected? True

```
-----------------------------------

R 5 | W 4 | B 1 | B 5 | G 3 | Y 6 |

R 1 | B 2 | Y 2 | G 2 | P 3 | B 3 |

P 6 | W 5 | R 6 | R 4 | G 6 | B 4 |

B 6 | G 5 | P 5 | W 2 | W 6 | G 1 |

P 2 | R 3 | P 1 | R 2 | P 4 | Y 3 |

W 1 | G 4 | W 3 | Y 1 | Y 4 | Y 5 |
```

Graph characteristics:
36 nodes
52 edges
Connected? True

```
-----------------------------------

R 4 | B 5 | P 4 | G 5 | Y 3 | R 3 |

Y 4 | B 3 | B 4 | Y 2 | R 6 | G 1 |

G 3 | B 1 | G 2 | W 6 | G 4 | Y 5 |

Y 6 | Y 1 | P 5 | R 1 | R 2 | P 2 |

P 6 | W 3 | B 6 | R 5 | W 1 | P 3 |
```

B 2 | G 6 | W 2 | P 1 | W 4 | W 5 |

Graph characteristics:
36 nodes
44 edges
Connected? False
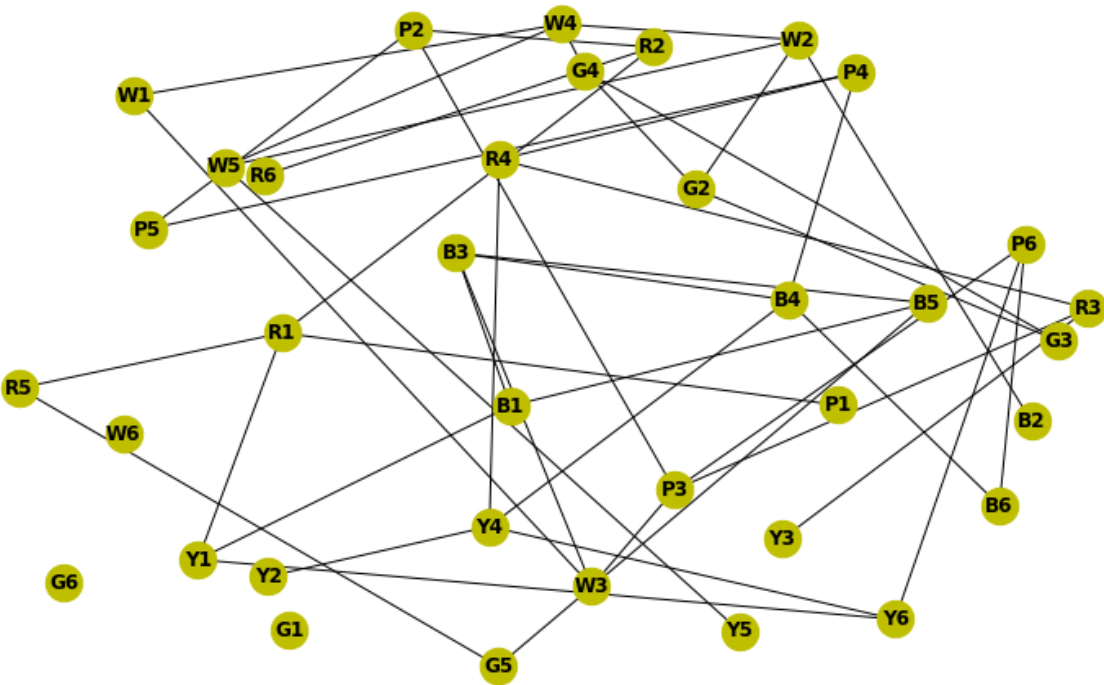
```
R4 --> ['Y4', 'P4', 'R3']
B5 --> ['B3', 'B1', 'G5']
P4 --> ['R4', 'B4', 'P5']
G5 --> ['B5', 'R5']
Y3 --> ['R3']
R3 --> ['R4', 'Y3', 'P3']
Y4 --> ['R4', 'Y6', 'B4', 'Y2']
B3 --> ['B5', 'B1', 'W3', 'B4']
B4 --> ['P4', 'Y4', 'B3', 'B6']
Y2 --> ['Y4']
R6 --> ['R2']
G1 --> []
G3 --> ['G2', 'G4']
B1 --> ['B5', 'B3', 'Y1']
G2 --> ['G3', 'W2', 'G4']
W6 --> []
G4 --> ['G3', 'G2', 'W4']
Y5 --> ['W5']
Y6 --> ['Y4', 'P6', 'Y1']
Y1 --> ['B1', 'Y6', 'R1']
P5 --> ['P4', 'P2']
R1 --> ['Y1', 'R5', 'P1', 'R2']
R2 --> ['R6', 'R1', 'P2']
P2 --> ['P5', 'R2', 'P3']
P6 --> ['Y6', 'B6', 'P3']
W3 --> ['B3', 'W1', 'P3']
B6 --> ['B4', 'P6']
R5 --> ['G5', 'R1']
W1 --> ['W3', 'W4']
P3 --> ['R3', 'P2', 'P6', 'W3']
B2 --> ['W2']
G6 --> []
W2 --> ['G2', 'B2', 'W4', 'W5']
P1 --> ['R1']
W4 --> ['G4', 'W1', 'W2', 'W5']
W5 --> ['Y5', 'W2', 'W4']
```

```
         ---------------------------------

Y 6 | R 5 | B 2 | G 2 | P 3 | W 3 |

B 1 | Y 5 | P 5 | B 5 | B 3 | G 5 |

B 4 | Y 2 | R 6 | Y 4 | P 1 | G 6 |

G 1 | R 1 | R 3 | P 6 | W 6 | Y 1 |

B 6 | R 4 | P 4 | W 2 | G 3 | G 4 |

Y 3 | P 2 | R 2 | W 5 | W 1 | W 4 |

Graph characteristics:
36 nodes
58 edges
Connected? True
```

```
----------------------------------

B 1 | G 6 | P 4 | W 1 | Y 6 | P 6 |

G 3 | P 5 | R 4 | B 2 | G 5 | G 4 |

W 4 | Y 2 | Y 5 | W 3 | Y 1 | P 2 |

P 3 | W 5 | Y 3 | Y 4 | B 5 | B 6 |

W 2 | G 2 | P 1 | B 4 | R 3 | R 1 |

B 3 | W 6 | R 2 | G 1 | R 5 | R 6 |
```

Graph characteristics:
36 nodes
54 edges
Connected? True

```
----------------------------------

G 2 | G 1 | W 3 | P 5 | P 3 | R 4 |

B 3 | P 4 | W 5 | B 1 | B 4 | G 4 |

W 4 | Y 5 | G 3 | Y 3 | B 2 | Y 4 |

R 3 | R 5 | P 1 | W 6 | R 6 | B 5 |

Y 1 | Y 6 | Y 2 | P 2 | B 6 | R 1 |

G 6 | P 6 | G 5 | W 2 | R 2 | W 1 |
```

Graph characteristics:
36 nodes
53 edges
Connected? False
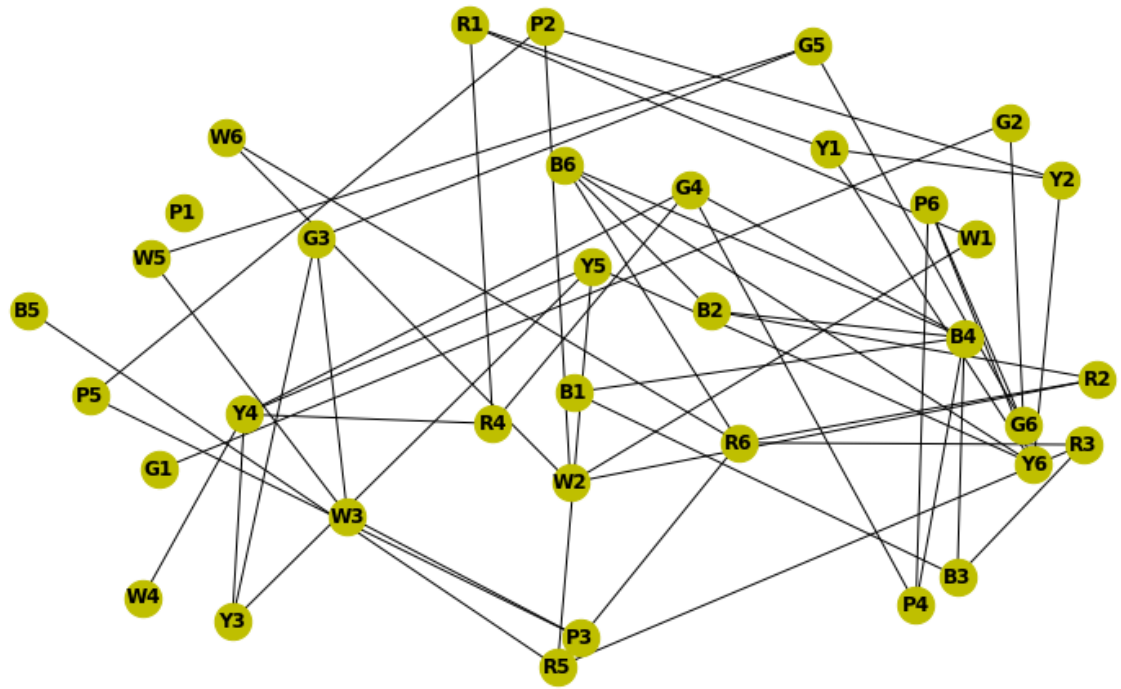
```
G2 --> ['G6', 'G1']
G1 --> ['G2']
W3 --> ['W5', 'G3', 'P3']
P5 --> ['P2', 'P3']
P3 --> ['W3', 'P5']
R4 --> ['G4', 'Y4', 'R1']
B3 --> ['R3', 'B1', 'B4']
P4 --> ['P6', 'B4', 'G4']
W5 --> ['W3', 'G5']
B1 --> ['B3', 'B4']
B4 --> ['B3', 'P4', 'B1', 'B2', 'B6', 'G4']
G4 --> ['R4', 'P4', 'B4', 'Y4']
W4 --> ['Y4']
Y5 --> ['R5', 'Y6', 'Y3', 'Y4']
G3 --> ['W3', 'G5', 'Y3']
Y3 --> ['Y5', 'G3', 'Y4']
B2 --> ['B4', 'B6', 'R2']
Y4 --> ['R4', 'G4', 'W4', 'Y5', 'Y3']
R3 --> ['B3', 'R5', 'R6']
R5 --> ['Y5', 'R3', 'R6', 'B5']
P1 --> []
W6 --> ['W2', 'R6']
R6 --> ['R3', 'R5', 'W6', 'B6', 'R2']
B5 --> ['R5']
Y1 --> ['Y6', 'Y2', 'R1']
Y6 --> ['Y5', 'Y1', 'P6', 'Y2', 'B6']
Y2 --> ['Y1', 'Y6', 'P2']
P2 --> ['P5', 'Y2', 'W2']
B6 --> ['B4', 'B2', 'R6', 'Y6']
R1 --> ['R4', 'Y1', 'W1']
G6 --> ['G2', 'P6', 'G5']
P6 --> ['P4', 'Y6', 'G6']
G5 --> ['W5', 'G3', 'G6']
W2 --> ['W6', 'P2', 'R2', 'W1']
R2 --> ['B2', 'R6', 'W2']
W1 --> ['R1', 'W2']
```

```
        ---------------------------------
```

```
Y 2 | W 2 | Y 6 | B 3 | P 6 | P 2 |

B 4 | G 4 | P 3 | R 3 | B 1 | W 1 |

W 5 | R 2 | P 1 | R 1 | P 5 | W 4 |

Y 3 | B 5 | W 3 | Y 1 | R 6 | B 2 |

B 6 | G 2 | Y 5 | G 6 | G 1 | W 6 |

Y 4 | R 4 | R 5 | G 5 | P 4 | G 3 |
```
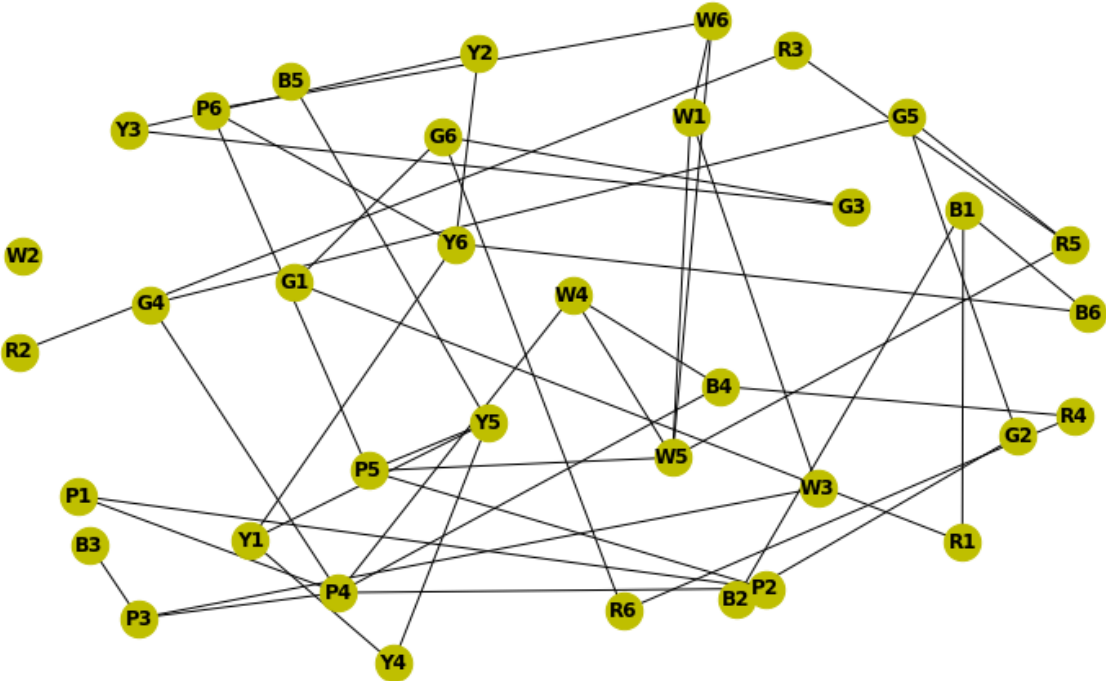
```
Graph characteristics:
36 nodes
58 edges
Connected? True
```

```
----------------------------------
```

W 6 | Y 3 | W 1 | P 5 | W 5 | P 6 |

B 3 | G 6 | P 3 | R 1 | G 1 | W 2 |

P 1 | G 3 | P 4 | P 2 | W 4 | B 4 |

G 5 | Y 2 | G 4 | B 6 | R 5 | Y 6 |

G 2 | R 6 | W 3 | B 1 | B 2 | R 4 |

Y 4 | B 5 | R 2 | Y 5 | R 3 | Y 1 |

Graph characteristics:
36 nodes
46 edges
Connected? False

```
W6 --> ['W1', 'W5', 'P6']
Y3 --> ['G3', 'Y2']
W1 --> ['W6', 'W3', 'W5']
P5 --> ['P2', 'Y5', 'W5', 'P6']
W5 --> ['W6', 'W1', 'P5', 'W4', 'R5']
P6 --> ['W6', 'P5', 'Y6']
B3 --> ['P3']
G6 --> ['G3', 'R6', 'G1']
P3 --> ['B3', 'P4', 'W3']
R1 --> ['B1', 'G1']
G1 --> ['G6', 'R1']
W2 --> []
P1 --> ['P4', 'P2']
G3 --> ['Y3', 'G6']
P4 --> ['P3', 'P1', 'G4', 'P2', 'W4', 'B4']
P2 --> ['P5', 'P1', 'P4']
W4 --> ['W5', 'P4', 'B4']
B4 --> ['P4', 'W4', 'R4']
G5 --> ['G2', 'G4', 'R5']
Y2 --> ['Y3', 'Y6']
G4 --> ['P4', 'G5']
B6 --> ['B1', 'Y6']
R5 --> ['W5', 'G5', 'R3']
Y6 --> ['P6', 'Y2', 'B6', 'Y1']
G2 --> ['G5', 'B2']
R6 --> ['G6', 'R4']
W3 --> ['W1', 'P3']
B1 --> ['R1', 'B6', 'B2']
B2 --> ['G2', 'B1']
R4 --> ['B4', 'R6']
Y4 --> ['Y5', 'Y1']
B5 --> ['Y5']
R2 --> ['R3']
Y5 --> ['P5', 'Y4', 'B5', 'Y1']
R3 --> ['R5', 'R2']
Y1 --> ['Y6', 'Y4', 'Y5']
```

```
W 3 | W 2 | R 1 | G 3 | B 5 | W 4 |

W 6 | W 1 | R 2 | P 6 | Y 3 | P 2 |

G 2 | G 6 | P 5 | Y 5 | W 5 | G 1 |

P 4 | R 3 | B 3 | G 4 | R 4 | B 4 |

R 6 | P 3 | Y 4 | B 6 | B 1 | Y 6 |

P 1 | Y 1 | G 5 | B 2 | Y 2 | R 5 |
```
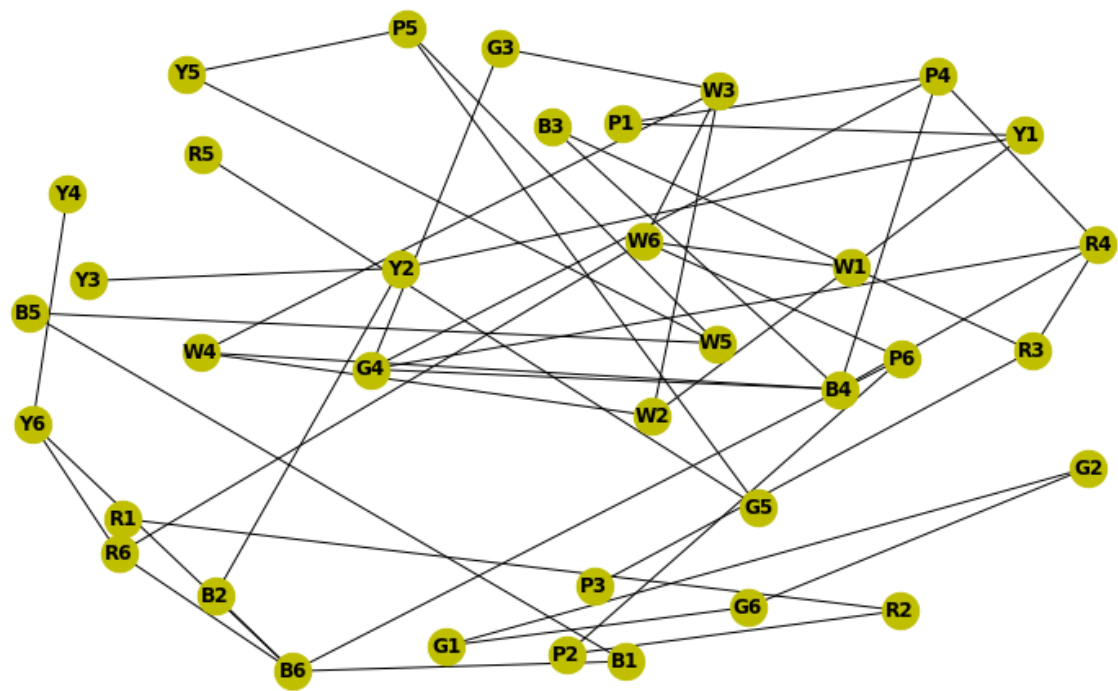
Graph characteristics:
36 nodes
47 edges
Connected? False

```
W3 --> ['W6', 'W2', 'G3', 'W4']
W2 --> ['W3', 'W1', 'W4']
R1 --> ['R2']
G3 --> ['W3', 'G4']
B5 --> ['W5', 'B1']
W4 --> ['W3', 'W2', 'B4']
W6 --> ['W3', 'R6', 'W1', 'P6']
W1 --> ['W2', 'W6', 'Y1']
R2 --> ['R1', 'P2']
P6 --> ['W6', 'B6', 'P2']
Y3 --> ['Y2']
P2 --> ['R2', 'P6']
G2 --> ['G6', 'G1']
G6 --> ['G2', 'G1']
P5 --> ['G5', 'Y5', 'W5']
Y5 --> ['P5', 'W5']
W5 --> ['B5', 'P5', 'Y5']
G1 --> ['G2', 'G6']
P4 --> ['P1', 'G4', 'R4', 'B4']
R3 --> ['P3', 'B3', 'R4']
B3 --> ['R3', 'B4']
G4 --> ['G3', 'P4', 'R4', 'B4']
R4 --> ['P4', 'R3', 'G4', 'B4']
B4 --> ['W4', 'P4', 'B3', 'G4', 'R4']
R6 --> ['W6', 'B6', 'Y6']
P3 --> ['R3']
Y4 --> ['Y6']
B6 --> ['P6', 'R6', 'B2', 'B1', 'Y6']
B1 --> ['B5', 'B6']
Y6 --> ['R6', 'Y4', 'B6']
P1 --> ['P4', 'Y1']
Y1 --> ['W1', 'P1', 'Y2']
G5 --> ['P5', 'R5']
B2 --> ['B6', 'Y2']
Y2 --> ['Y3', 'Y1', 'B2']
R5 --> ['G5']
```

```
          ---------------------------------


R 3 | R 4 | G 2 | W 3 | W 5 | P 4 |

Y 3 | R 1 | P 1 | W 6 | W 1 | R 5 |

R 6 | B 6 | G 6 | P 2 | P 6 | B 5 |

P 3 | W 4 | R 2 | G 3 | B 4 | B 1 |

Y 1 | P 5 | W 2 | B 2 | B 3 | G 4 |

G 1 | Y 4 | G 5 | Y 5 | Y 2 | Y 6 |
```
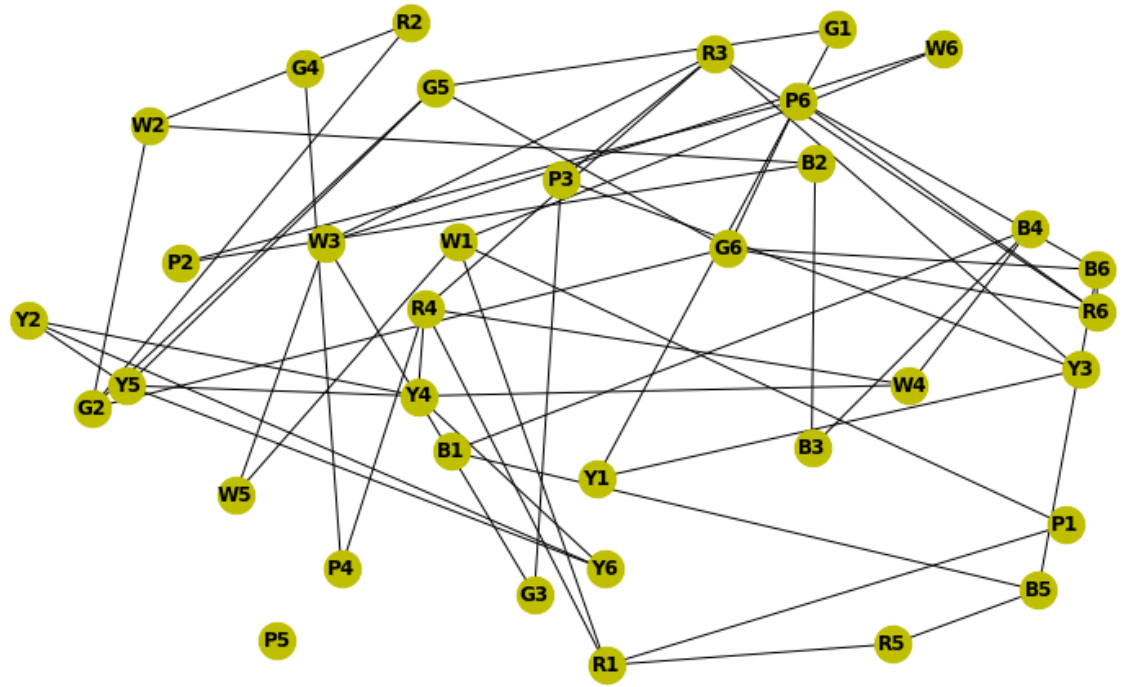
```
Graph characteristics:
36 nodes
54 edges
Connected? False
```

```
R3 --> ['Y3', 'R6', 'P3', 'R4', 'W3']
R4 --> ['R3', 'R1', 'W4', 'Y4', 'P4']
G2 --> ['G6', 'R2', 'W2', 'G5']
W3 --> ['R3', 'W6', 'G3', 'W5']
W5 --> ['W3', 'W1']
P4 --> ['R4', 'G4']
Y3 --> ['R3', 'P3', 'Y1']
R1 --> ['R4', 'P1', 'W1', 'R5']
P1 --> ['R1', 'W1']
W6 --> ['W3', 'W1']
W1 --> ['W5', 'R1', 'P1', 'W6']
R5 --> ['R1', 'B5']
R6 --> ['R3', 'B6', 'G6', 'P6']
B6 --> ['R6', 'G6', 'P6', 'B5']
G6 --> ['G2', 'R6', 'B6', 'G5', 'P6']
P2 --> ['B2', 'P6']
P6 --> ['R6', 'B6', 'G6', 'P2']
B5 --> ['R5', 'B6', 'B1']
P3 --> ['R3', 'Y3', 'G3']
W4 --> ['R4', 'Y4', 'B4']
R2 --> ['G2', 'W2']
G3 --> ['W3', 'P3']
B4 --> ['W4', 'B3', 'B1']
B1 --> ['B5', 'B4']
Y1 --> ['Y3', 'G1']
P5 --> []
W2 --> ['G2', 'R2', 'B2']
B2 --> ['P2', 'W2', 'B3']
B3 --> ['B4', 'B2']
G4 --> ['P4']
G1 --> ['Y1', 'G5']
Y4 --> ['R4', 'W4', 'Y5', 'Y2', 'Y6']
G5 --> ['G2', 'G6', 'G1', 'Y5']
Y5 --> ['Y4', 'G5', 'Y2', 'Y6']
Y2 --> ['Y4', 'Y5', 'Y6']
Y6 --> ['Y4', 'Y5', 'Y2']
```

```
          ---------------------------------


R 5 | Y 5 | Y 6 | W 3 | Y 3 | W 6 |

B 4 | P 2 | W 1 | P 3 | Y 1 | R 2 |

B 5 | W 2 | B 1 | W 5 | G 4 | R 1 |

R 4 | P 4 | Y 2 | G 3 | R 6 | P 6 |

R 3 | B 2 | B 3 | G 2 | P 1 | G 6 |

B 6 | W 4 | Y 4 | G 1 | P 5 | G 5 |

Graph characteristics:
36 nodes
59 edges
Connected? False
```
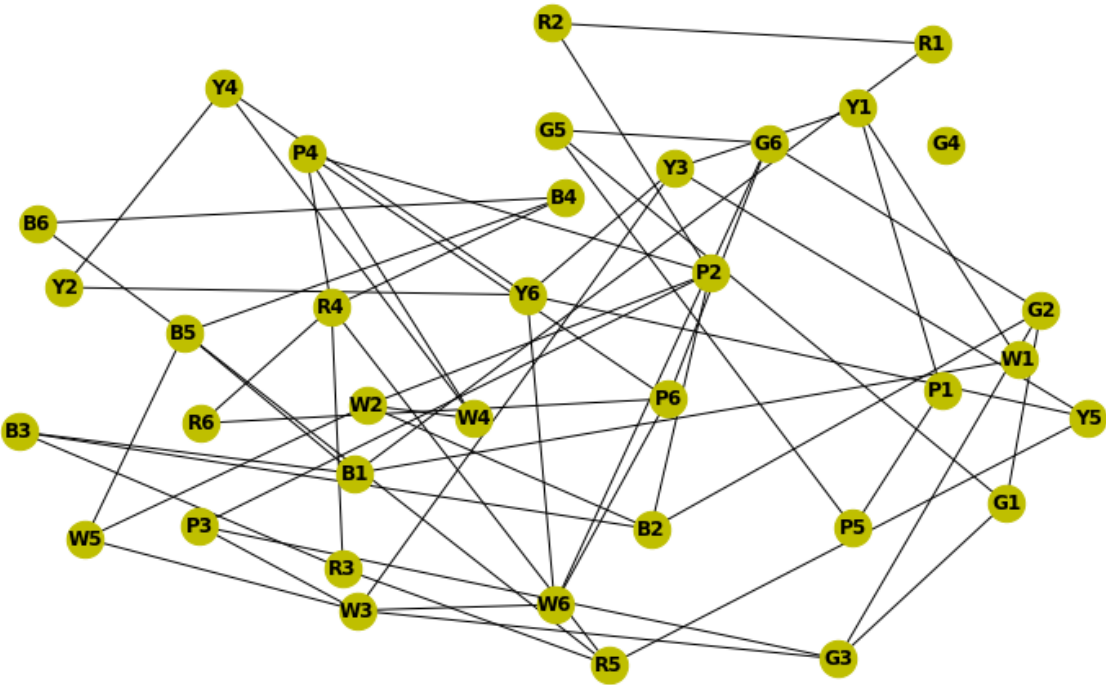
```
R5 --> ['B5', 'R4', 'R3', 'Y5']
Y5 --> ['R5', 'Y6', 'Y3']
Y6 --> ['Y5', 'Y2', 'Y4', 'Y3', 'W6']
W3 --> ['P3', 'W5', 'G3', 'Y3', 'W6']
Y3 --> ['Y5', 'Y6', 'W3', 'Y1']
W6 --> ['Y6', 'W3', 'P6', 'G6']
B4 --> ['B5', 'R4', 'B6']
P2 --> ['W2', 'P4', 'B2', 'P3', 'R2']
W1 --> ['B1', 'Y1']
P3 --> ['W3', 'P2', 'G3']
Y1 --> ['Y3', 'W1', 'P1']
R2 --> ['P2', 'R1']
B5 --> ['R5', 'B4', 'B6', 'B1', 'W5']
W2 --> ['P2', 'B2', 'W4', 'W5']
B1 --> ['W1', 'B5', 'B3', 'R1']
W5 --> ['W3', 'B5', 'W2']
G4 --> []
R1 --> ['R2', 'B1']
R4 --> ['R5', 'B4', 'R3', 'P4', 'R6']
P4 --> ['P2', 'R4', 'W4', 'P6']
Y2 --> ['Y6', 'Y4']
G3 --> ['W3', 'P3', 'G2', 'G1']
R6 --> ['R4', 'P6']
P6 --> ['W6', 'P4', 'R6', 'G6']
R3 --> ['R5', 'R4', 'B3']
B2 --> ['P2', 'W2', 'B3', 'G2']
B3 --> ['B1', 'R3', 'B2']
G2 --> ['G3', 'B2', 'G1', 'G6']
P1 --> ['Y1', 'P5']
G6 --> ['W6', 'P6', 'G2', 'G5']
B6 --> ['B4', 'B5']
W4 --> ['W2', 'P4', 'Y4']
Y4 --> ['Y6', 'Y2', 'W4']
G1 --> ['G3', 'G2', 'G5']
P5 --> ['P1', 'G5']
G5 --> ['G6', 'G1', 'P5']
```

```
       ---------------------------------
```

```
P 1 | Y 4 | G 5 | B 6 | Y 2 | Y 5 |

B 3 | B 1 | W 4 | G 3 | B 5 | W 6 |

Y 3 | G 4 | W 2 | R 4 | B 2 | G 1 |

W 1 | Y 1 | W 3 | R 5 | G 6 | R 3 |

W 5 | R 2 | R 1 | P 4 | G 2 | R 6 |

B 4 | Y 6 | P 5 | P 6 | P 3 | P 2 |
```
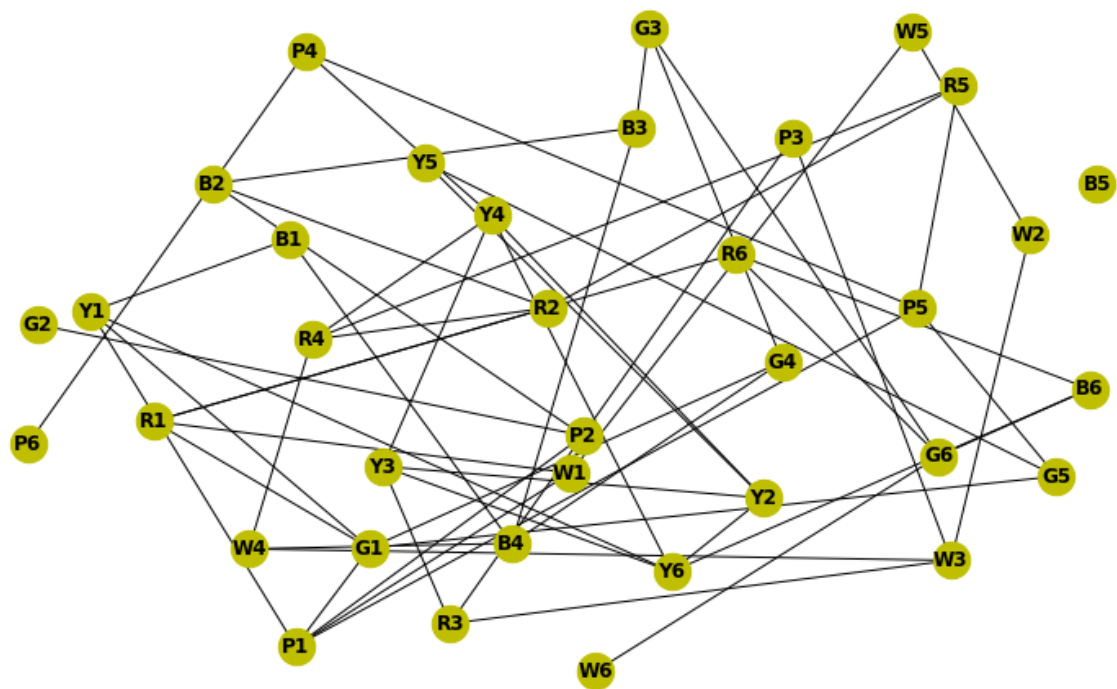
```
Graph characteristics:
36 nodes
51 edges
Connected? True
```

```
-----------------------------------

R 1 | W 1 | P 3 | B 6 | G 6 | R 6 |

R 2 | B 2 | W 3 | R 4 | R 5 | W 4 |

B 5 | P 2 | R 3 | Y 3 | W 6 | G 2 |

P 6 | W 5 | W 2 | Y 4 | P 4 | B 1 |

G 4 | B 3 | Y 5 | Y 2 | G 3 | B 4 |

G 1 | P 1 | G 5 | Y 6 | P 5 | Y 1 |
```

Graph characteristics:
36 nodes
56 edges
Connected? False

```
R1 --> ['R2', 'G1', 'W1', 'R6']
W1 --> ['R1', 'W5', 'P1']
P3 --> ['W3', 'R3']
B6 --> ['Y6', 'G6', 'R6']
G6 --> ['B6', 'W6', 'G3', 'R6']
R6 --> ['R1', 'B6', 'G6']
R2 --> ['R1', 'B2', 'R4', 'R5']
B2 --> ['R2', 'P2', 'B3']
W3 --> ['P3', 'R3', 'W2', 'W4']
R4 --> ['R2', 'Y4', 'R5', 'W4']
R5 --> ['R2', 'R4', 'P5']
W4 --> ['W3', 'R4', 'B4']
B5 --> []
P2 --> ['B2', 'P1', 'G2']
R3 --> ['P3', 'W3', 'Y3']
Y3 --> ['R3', 'Y4', 'Y2', 'Y6']
W6 --> ['G6']
G2 --> ['P2']
P6 --> ['P4']
W5 --> ['W1', 'W2']
W2 --> ['W3', 'W5']
Y4 --> ['R4', 'Y3', 'Y2', 'Y6', 'P4']
P4 --> ['P6', 'Y4', 'P5']
B1 --> ['B4', 'Y1']
G4 --> ['G1', 'G3', 'B4']
B3 --> ['B2', 'G3', 'B4']
Y5 --> ['G5', 'Y2']
Y2 --> ['Y3', 'Y4', 'Y5', 'Y6']
G3 --> ['G6', 'G4', 'B3']
B4 --> ['W4', 'B1', 'G4', 'B3']
G1 --> ['R1', 'G4', 'P1', 'G5', 'Y1']
P1 --> ['W1', 'P2', 'G1', 'P5', 'Y1']
G5 --> ['Y5', 'G1', 'P5']
Y6 --> ['B6', 'Y3', 'Y4', 'Y2', 'Y1']
P5 --> ['R5', 'P4', 'P1', 'G5']
Y1 --> ['B1', 'G1', 'P1', 'Y6']
```

```
         ---------------------------------

P 6 | W 3 | B 6 | G 1 | P 4 | W 2 |

P 2 | W 6 | P 3 | R 2 | R 6 | W 5 |

Y 5 | P 1 | G 5 | P 5 | Y 1 | G 2 |

B 4 | W 1 | B 5 | R 4 | R 5 | W 4 |

B 3 | B 2 | R 3 | Y 6 | Y 2 | G 4 |

Y 3 | R 1 | B 1 | G 3 | Y 4 | G 6 |
```

```
Graph characteristics:
36 nodes
60 edges
Connected? True
```

```
          ----------------------------------

          P 5 | B 3 | W 2 | B 5 | G 6 | W 4 |

          P 4 | R 1 | B 4 | W 3 | R 4 | G 2 |

          W 5 | P 2 | R 5 | Y 4 | W 1 | W 6 |

          B 2 | B 6 | P 3 | Y 3 | Y 2 | R 3 |

          Y 6 | R 2 | P 6 | G 1 | Y 5 | R 6 |

          B 1 | Y 1 | P 1 | G 4 | G 3 | G 5 |
```

Graph characteristics:
36 nodes
48 edges
Connected? True


```
          ----------------------------------

          G 5 | Y 5 | P 2 | B 6 | Y 4 | G 1 |

          W 5 | B 5 | P 3 | P 4 | W 4 | B 2 |

          W 2 | Y 3 | Y 1 | R 2 | G 6 | G 2 |

          B 3 | W 3 | Y 6 | P 6 | R 1 | R 5 |

          Y 2 | W 6 | R 3 | B 4 | B 1 | P 1 |

          R 4 | W 1 | G 3 | R 6 | P 5 | G 4 |
```

Graph characteristics:
36 nodes
49 edges
Connected? False
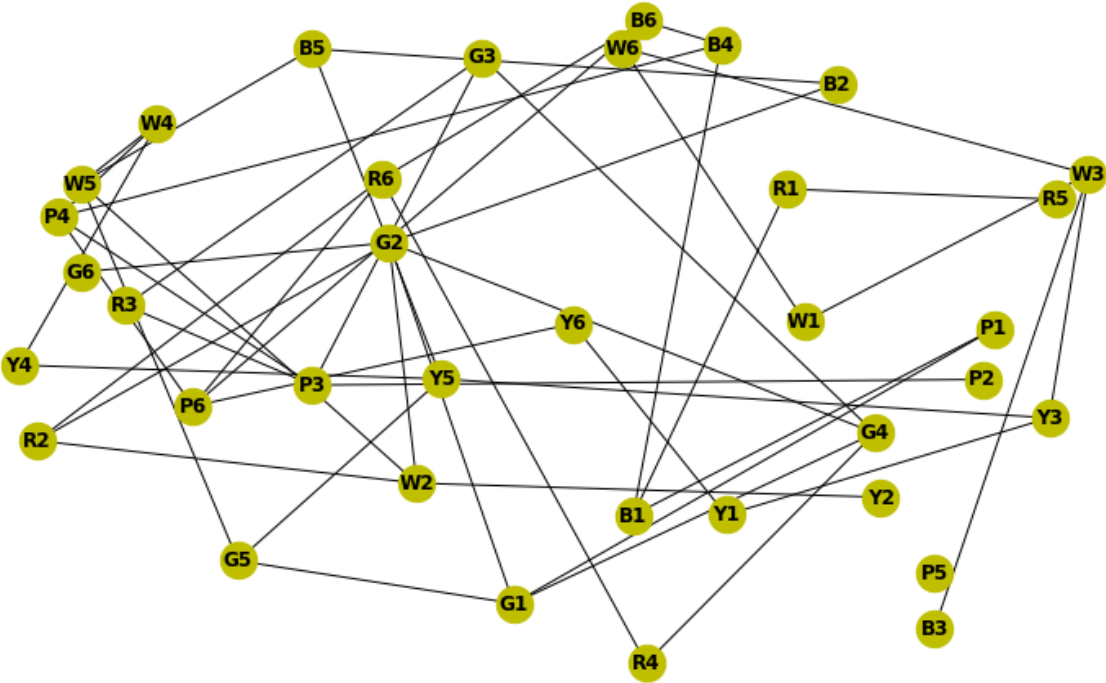
```
G5 --> ['W5', 'Y5', 'G1']
Y5 --> ['G5', 'B5', 'Y3', 'Y4']
P2 --> ['P3']
B6 --> ['P6', 'B4', 'R6']
Y4 --> ['Y5', 'W4']
G1 --> ['G5', 'G2', 'P1', 'G4']
W5 --> ['G5', 'W2', 'B5', 'W4']
B5 --> ['Y5', 'W5', 'B2']
P3 --> ['P2', 'R3', 'G3', 'P4']
P4 --> ['P3', 'P6', 'B4', 'W4']
W4 --> ['Y4', 'W5', 'P4']
B2 --> ['B5', 'G2']
W2 --> ['W5', 'Y2', 'R2', 'G2']
Y3 --> ['Y5', 'W3', 'Y1']
Y1 --> ['Y3', 'Y6']
R2 --> ['W2', 'R6', 'G2']
G6 --> ['G2']
G2 --> ['G1', 'B2', 'W2', 'R2', 'G6', 'G4']
B3 --> ['W3']
W3 --> ['Y3', 'B3', 'W6', 'W1']
Y6 --> ['Y1', 'P6']
P6 --> ['B6', 'P4', 'Y6', 'R6']
R1 --> ['B1', 'R5']
R5 --> ['R1']
Y2 --> ['W2']
W6 --> ['W3', 'W1']
R3 --> ['P3', 'G3']
B4 --> ['B6', 'P4', 'B1']
B1 --> ['R1', 'B4', 'P1']
P1 --> ['G1', 'B1']
R4 --> ['R6', 'G4']
W1 --> ['W3', 'W6']
G3 --> ['P3', 'R3', 'G4']
R6 --> ['B6', 'R2', 'P6', 'R4']
P5 --> []
G4 --> ['G1', 'G2', 'R4', 'G3']
```

B 3 | G 1 | G 3 | G 2 | B 5 | R 6 |

G 4 | P 3 | B 2 | B 1 | R 5 | B 6 |

B 4 | P 2 | P 5 | R 2 | G 6 | W 5 |

W 6 | G 5 | R 3 | Y 3 | W 2 | R 1 |

R 4 | Y 1 | W 3 | P 6 | P 1 | W 4 |
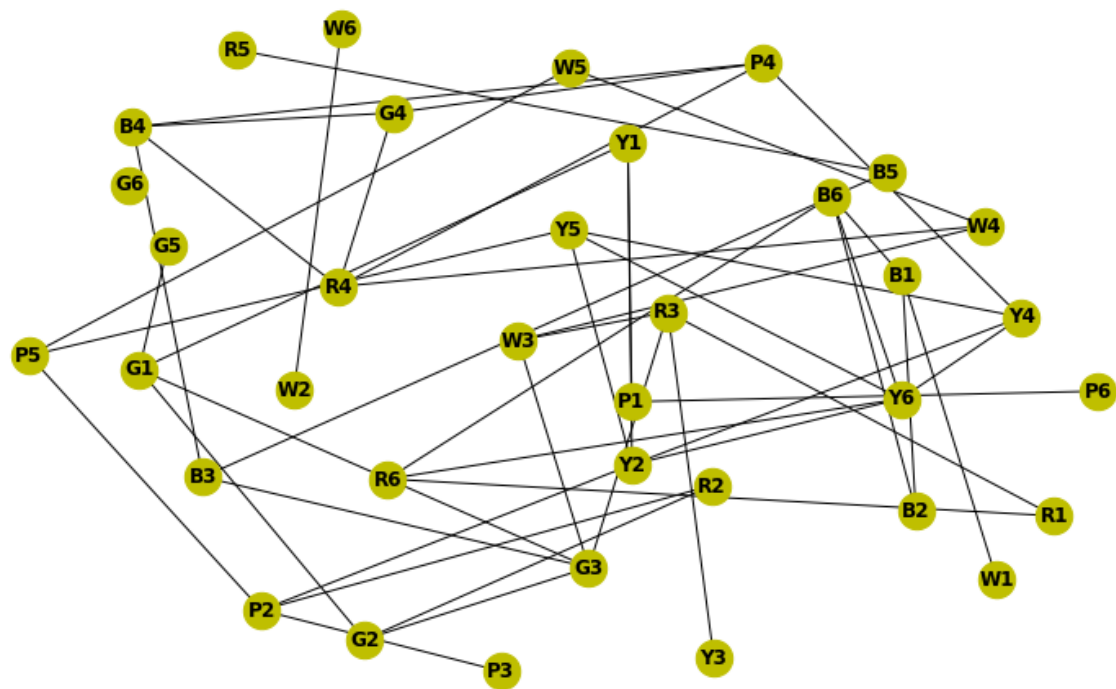
P 4 | Y 2 | Y 5 | W 1 | Y 4 | Y 6 |

Graph characteristics:
36 nodes
49 edges
Connected? False

```
            B3 --> ['B4', 'G3', 'B5']
            G1 --> ['G5', 'Y1', 'G3', 'G2']
            G3 --> ['B3', 'G1', 'R3', 'W3', 'G2']
            G2 --> ['G1', 'G3', 'R2']
            B5 --> ['B3', 'R5']
            R6 --> ['B6', 'R1', 'Y6']
            G4 --> ['B4', 'R4', 'P4']
            P3 --> ['P2']
            B2 --> ['B1', 'B6']
            B1 --> ['B2', 'W1', 'B6']
            R5 --> ['B5']
            B6 --> ['R6', 'B2', 'B1', 'Y6']
            B4 --> ['B3', 'G4', 'R4', 'P4']
            P2 --> ['P3', 'Y2', 'P5', 'R2']
            P5 --> ['P2', 'Y5', 'W5']
            R2 --> ['G2', 'P2']
            G6 --> []
            W5 --> ['P5', 'W4']
            W6 --> ['W2']
            G5 --> ['G1']
            R3 --> ['G3', 'W3', 'Y3', 'R1']
            Y3 --> ['R3']
            W2 --> ['W6']
            R1 --> ['R6', 'R3']
            R4 --> ['G4', 'B4', 'P4', 'W4']
            Y1 --> ['G1', 'Y2', 'P1']
            W3 --> ['G3', 'R3', 'W4']
            P6 --> ['P1']
            P1 --> ['Y1', 'P6']
            W4 --> ['W5', 'R4', 'W3']
            P4 --> ['G4', 'B4', 'R4', 'Y4']
            Y2 --> ['P2', 'Y1', 'Y5', 'Y4', 'Y6']
            Y5 --> ['P5', 'Y2', 'Y4', 'Y6']
            W1 --> ['B1']
            Y4 --> ['P4', 'Y2', 'Y5', 'Y6']
            Y6 --> ['R6', 'B6', 'Y2', 'Y5', 'Y4']


            ---------------------------------


            W 6 | G 5 | B 1 | G 4 | P 6 | Y 3 |

            W 4 | P 4 | G 6 | W 5 | B 6 | Y 2 |

            P 3 | G 2 | R 4 | G 3 | B 5 | B 3 |

            R 5 | P 5 | P 1 | G 1 | R 3 | P 2 |

            Y 5 | Y 6 | W 3 | B 2 | Y 1 | R 2 |

            W 1 | W 2 | R 6 | Y 4 | R 1 | B 4 |

            Graph characteristics:
            36 nodes
            49 edges
            Connected? False
```
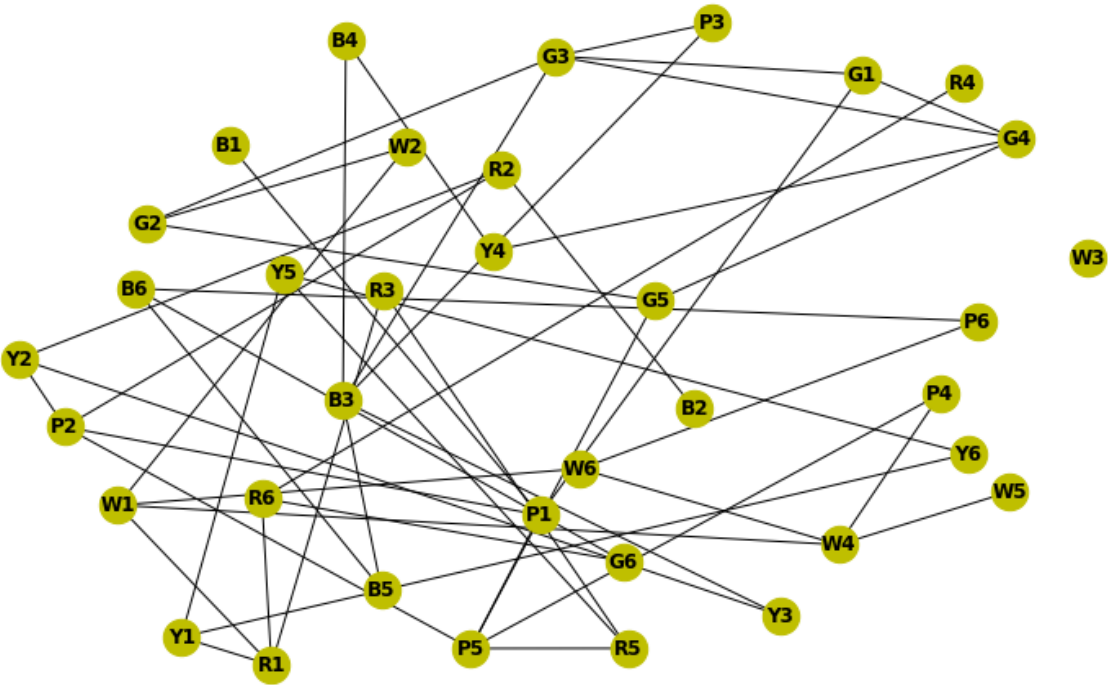
```
        W6 --> ['W4', 'W1', 'P6']
        G5 --> ['G2', 'P5', 'G4']
        B1 --> ['P1']
        G4 --> ['G5', 'G3', 'G1', 'Y4']
        P6 --> ['W6', 'B6']
        Y3 --> ['Y2', 'B3']
        W4 --> ['W6', 'W1', 'P4', 'W5']
        P4 --> ['W4', 'P5']
        G6 --> ['R6', 'B6']
        W5 --> ['W4']
        B6 --> ['P6', 'G6', 'B5']
        Y2 --> ['Y3', 'P2', 'R2']
        P3 --> ['G3', 'B3']
        G2 --> ['G5', 'W2', 'G3']
        R4 --> ['R6']
        G3 --> ['G4', 'P3', 'G2', 'G1', 'B3']
        B5 --> ['B6', 'B3']
        B3 --> ['Y3', 'P3', 'G3', 'B5', 'B4']
        R5 --> ['Y5', 'P5', 'R3']
        P5 --> ['G5', 'P4', 'R5', 'P1', 'P2']
        P1 --> ['B1', 'P5', 'G1', 'P2']
        G1 --> ['G4', 'G3', 'P1']
        R3 --> ['R5', 'R1']
        P2 --> ['Y2', 'P5', 'P1', 'R2']
        Y5 --> ['R5', 'Y6', 'Y1']
        Y6 --> ['Y5', 'Y1']
        W3 --> []
        B2 --> ['R2']
        Y1 --> ['Y5', 'Y6', 'R1']
        R2 --> ['Y2', 'P2', 'B2']
        W1 --> ['W6', 'W4', 'W2', 'R1']
        W2 --> ['G2', 'W1']
        R6 --> ['G6', 'R4', 'R1']
        Y4 --> ['G4', 'B4']
        R1 --> ['R3', 'Y1', 'W1', 'R6']
        B4 --> ['B3', 'Y4']


        ----------------------------------


        P 5 | P 6 | G 5 | R 4 | Y 3 | R 5 |

        B 4 | B 1 | Y 2 | B 3 | R 1 | P 1 |

        B 2 | Y 6 | R 2 | G 1 | W 4 | P 4 |

        W 2 | Y 1 | Y 5 | B 5 | B 6 | Y 4 |

        R 3 | P 2 | R 6 | W 5 | G 2 | G 6 |

        P 3 | G 3 | W 3 | G 4 | W 1 | W 6 |

        Graph characteristics:
        36 nodes
        50 edges
        Connected? False
```
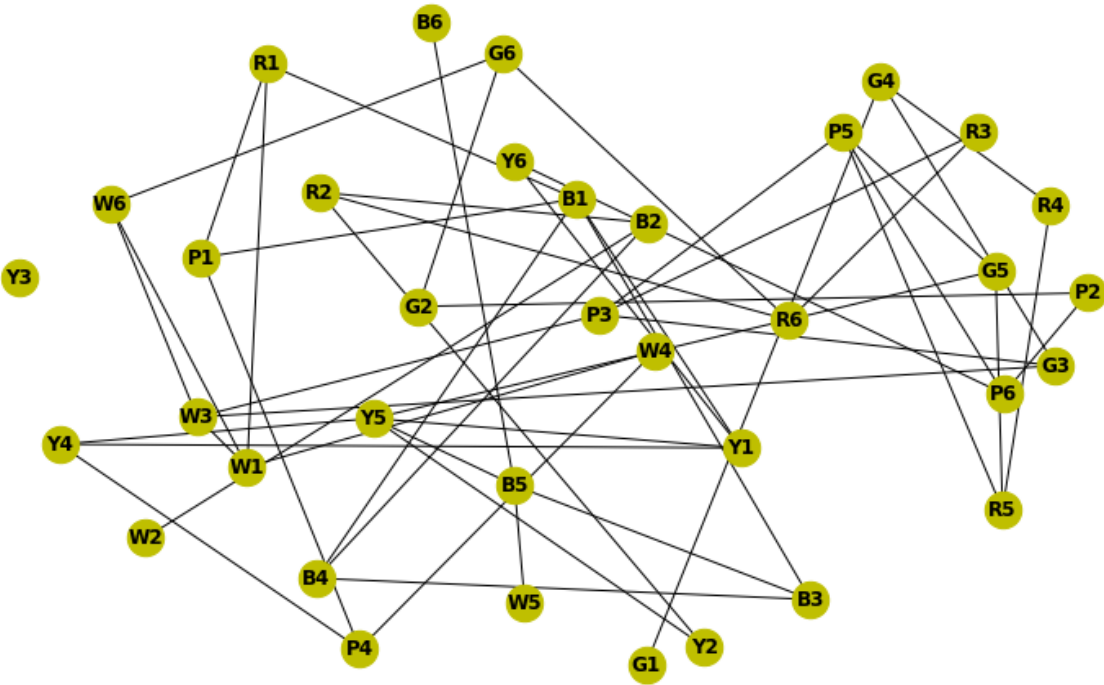
```
P5 --> ['P3', 'P6', 'G5', 'R5']
P6 --> ['P5', 'Y6', 'P2']
G5 --> ['P5', 'Y5', 'R5']
R4 --> ['G4', 'R5']
Y3 --> []
R5 --> ['P5', 'G5', 'R4']
B4 --> ['B2', 'B1', 'B3']
B1 --> ['B4', 'Y1', 'B3', 'R1', 'P1']
Y2 --> ['R2', 'Y5']
B3 --> ['B4', 'B1', 'B5']
R1 --> ['B1', 'W1', 'P1']
P1 --> ['B1', 'R1', 'P4']
B2 --> ['B4', 'W2', 'R2']
Y6 --> ['P6', 'Y1']
R2 --> ['Y2', 'B2', 'R6']
G1 --> ['G4']
W4 --> ['W1', 'P4']
P4 --> ['P1', 'W4', 'Y4']
W2 --> ['B2']
Y1 --> ['B1', 'Y6', 'Y5', 'Y4']
Y5 --> ['G5', 'Y2', 'Y1', 'B5', 'Y4']
B5 --> ['B3', 'Y5', 'W5', 'B6']
B6 --> ['B5']
Y4 --> ['P4', 'Y1', 'Y5']
R3 --> ['P3', 'R6']
P2 --> ['P6', 'G2']
R6 --> ['R2', 'R3', 'G6']
W5 --> ['B5']
G2 --> ['P2', 'G6']
G6 --> ['R6', 'G2', 'W6']
P3 --> ['P5', 'R3', 'G3', 'W3']
G3 --> ['P3', 'W3', 'G4']
W3 --> ['P3', 'G3', 'W1', 'W6']
G4 --> ['R4', 'G1', 'G3']
W1 --> ['R1', 'W4', 'W3', 'W6']
W6 --> ['G6', 'W3', 'W1']
```

```
R 2 | R 4 | R 6 | P 4 | G 1 | G 5 |

R 5 | W 2 | R 1 | B 3 | R 3 | P 2 |

W 6 | Y 4 | W 1 | P 1 | Y 2 | Y 5 |

B 1 | B 2 | W 5 | Y 6 | P 6 | Y 1 |

W 3 | W 4 | G 6 | Y 3 | B 4 | B 6 |

P 5 | P 3 | G 4 | G 2 | B 5 | G 3 |
```

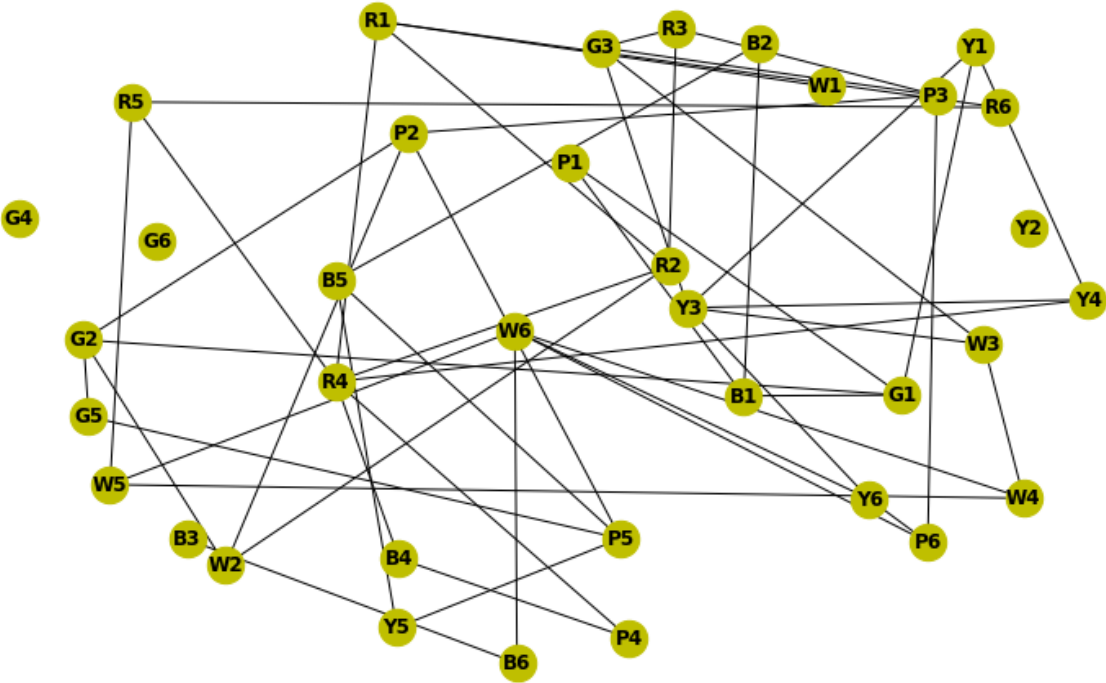Graph characteristics:
36 nodes
50 edges
Connected? True

```
----------------------------------
```

P 2 | B 1 | B 6 | B 3 | P 5 | P 3 |

Y 3 | W 5 | B 5 | R 4 | R 3 | R 5 |

G 2 | P 4 | B 2 | W 2 | P 1 | P 6 |

R 1 | G 4 | R 2 | Y 4 | G 1 | G 5 |

W 4 | B 4 | G 3 | W 3 | Y 6 | Y 2 |

R 6 | G 6 | Y 5 | W 1 | Y 1 | W 6 |

Graph characteristics:
36 nodes
61 edges
Connected? True

```
----------------------------------
```

R 2 | R 1 | P 6 | Y 4 | W 2 | R 4 |

G 1 | R 6 | W 5 | Y 1 | G 2 | R 5 |

Y 5 | B 5 | W 4 | W 3 | P 5 | G 6 |

P 1 | G 4 | W 6 | B 6 | B 3 | Y 2 |

B 1 | B 2 | Y 6 | Y 3 | G 5 | P 4 |

R 3 | W 1 | P 3 | G 3 | P 2 | B 4 |

Graph characteristics:
36 nodes
51 edges
Connected? False

```
R2 --> ['R3', 'R1', 'W2', 'R4']
R1 --> ['R2', 'R6', 'W1', 'R4']
P6 --> ['W6', 'Y6', 'P3']
Y4 --> ['Y1', 'Y3', 'R4']
W2 --> ['R2', 'G2', 'P2']
R4 --> ['R2', 'R1', 'Y4', 'R5', 'P4', 'B4']
G1 --> ['P1', 'B1', 'Y1', 'G2']
R6 --> ['R1', 'R5']
W5 --> ['W4', 'W6', 'R5']
Y1 --> ['Y4', 'G1', 'Y3']
G2 --> ['W2', 'G1', 'G5', 'P2']
R5 --> ['R4', 'R6', 'W5']
Y5 --> ['B5', 'P5']
B5 --> ['Y5', 'B2', 'P5']
W4 --> ['W5', 'W6', 'W3']
W3 --> ['W4', 'Y3', 'G3']
P5 --> ['Y5', 'B5', 'G5', 'P2']
G6 --> []
P1 --> ['G1', 'B1']
G4 --> []
W6 --> ['P6', 'W5', 'W4', 'Y6', 'B6']
B6 --> ['W6', 'B3']
B3 --> ['B6']
Y2 --> []
B1 --> ['G1', 'P1', 'B2']
B2 --> ['B5', 'B1']
Y6 --> ['P6', 'W6', 'Y3']
Y3 --> ['Y4', 'Y1', 'W3', 'Y6', 'G3']
G5 --> ['G2', 'P5']
P4 --> ['R4', 'B4']
R3 --> ['R2', 'P3', 'G3']
W1 --> ['R1']
P3 --> ['P6', 'R3', 'G3', 'P2']
G3 --> ['W3', 'Y3', 'R3', 'P3']
P2 --> ['W2', 'G2', 'P5', 'P3']
B4 --> ['R4', 'P4']
```

```
P 5 | P 2 | W 3 | B 1 | R 6 | P 1 |

R 5 | Y 5 | G 5 | G 2 | Y 6 | W 1 |

R 2 | W 2 | Y 3 | G 3 | P 3 | Y 1 |

R 3 | R 1 | Y 4 | G 4 | W 5 | B 5 |

W 4 | R 4 | Y 2 | W 6 | B 3 | G 1 |

P 4 | P 6 | G 6 | B 4 | B 6 | B 2 |
```

Graph characteristics:
36 nodes
58 edges
Connected? True

B 3 | W 1 | P 5 | Y 4 | W 2 | W 3 |

R 4 | P 1 | W 6 | R 2 | R 6 | W 4 |

W 5 | R 5 | Y 2 | Y 3 | B 5 | Y 6 |

R 1 | B 6 | R 3 | B 4 | P 6 | P 2 |

B 2 | B 1 | P 3 | P 4 | G 3 | G 5 |

G 2 | Y 5 | Y 1 | G 4 | G 6 | G 1 |

Graph characteristics:
36 nodes
56 edges
Connected? True

Y 6 | Y 2 | R 1 | G 6 | G 1 | R 6 |

G 2 | B 2 | P 5 | R 5 | R 3 | R 2 |

B 3 | B 5 | W 1 | Y 3 | R 4 | W 4 |

B 4 | P 2 | P 3 | G 5 | Y 1 | W 2 |

B 6 | G 4 | G 3 | W 3 | W 6 | W 5 |

B 1 | Y 4 | Y 5 | P 1 | P 4 | P 6 |

Graph characteristics:
36 nodes
61 edges
Connected? True

B 4 | Y 1 | Y 4 | B 5 | P 6 | B 2 |

B 6 | Y 2 | Y 6 | Y 3 | Y 5 | R 1 |

W 6 | B 1 | G 3 | P 2 | P 1 | R 6 |

P 3 | G 4 | G 2 | P 5 | R 3 | W 1 |

W 5 | W 3 | R 2 | W 4 | W 2 | R 5 |

B 3 | G 5 | P 4 | R 4 | G 1 | G 6 |

Graph characteristics:
36 nodes
54 edges
Connected? True

Y 2 | P 2 | P 5 | W 1 | B 6 | R 1 |

B 2 | Y 6 | R 3 | B 4 | B 1 | P 4 |

R 4 | B 5 | W 6 | P 3 | G 5 | G 4 |

R 6 | R 2 | P 1 | B 3 | W 3 | G 6 |

W 5 | R 5 | W 4 | G 3 | G 1 | Y 4 |

Y 1 | W 2 | Y 3 | Y 5 | G 2 | P 6 |

Graph characteristics:
36 nodes
47 edges
Connected? False
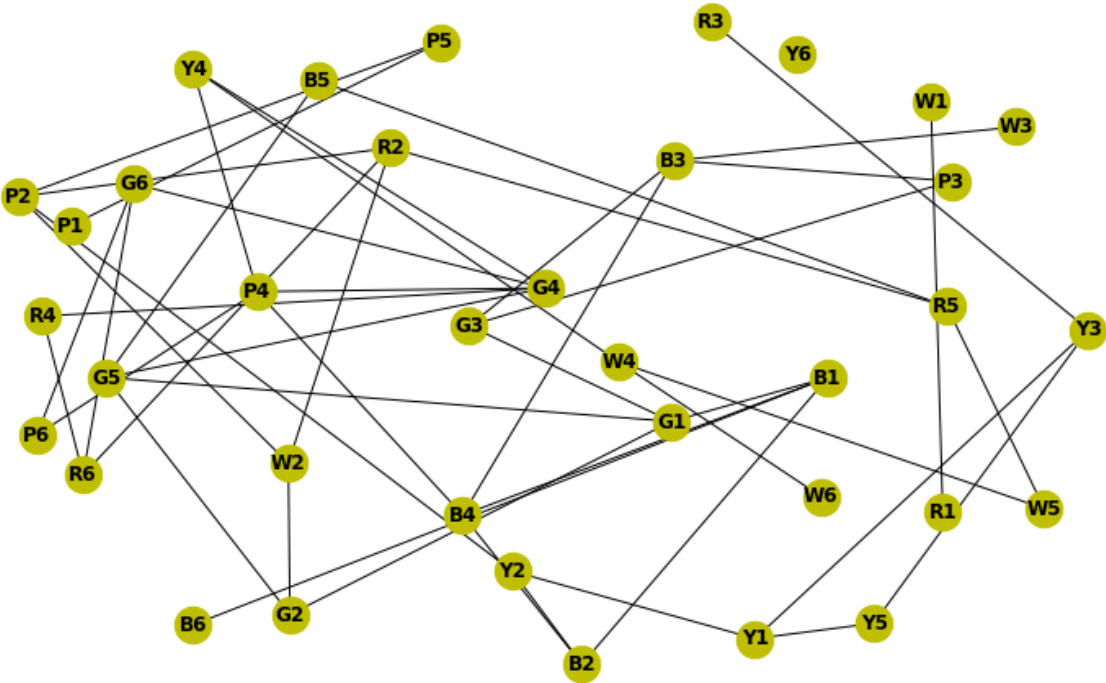
```
Y2 --> ['B2', 'Y1', 'P2']
P2 --> ['Y2', 'R2', 'W2', 'P5']
P5 --> ['P2', 'P1']
W1 --> ['R1']
B6 --> ['B1']
R1 --> ['W1']
B2 --> ['Y2', 'B4', 'B1']
Y6 --> []
R3 --> ['Y3']
B4 --> ['B2', 'B3', 'B1', 'P4']
B1 --> ['B6', 'B2', 'B4', 'G1']
P4 --> ['B4', 'G4', 'Y4', 'P6']
R4 --> ['R6', 'G4']
B5 --> ['R5', 'G5']
W6 --> ['W4']
P3 --> ['B3', 'G3']
G5 --> ['B5', 'G1', 'G2', 'G4']
G4 --> ['P4', 'R4', 'G5', 'G6', 'Y4']
R6 --> ['R4', 'R2', 'G6']
R2 --> ['P2', 'R6', 'R5', 'W2']
P1 --> ['P5']
B3 --> ['B4', 'P3', 'G3', 'W3']
W3 --> ['B3']
G6 --> ['G4', 'R6', 'P6']
W5 --> ['R5', 'W4']
R5 --> ['B5', 'R2', 'W5']
W4 --> ['W6', 'W5', 'Y4']
G3 --> ['P3', 'B3', 'G1']
G1 --> ['B1', 'G5', 'G3', 'G2']
Y4 --> ['P4', 'G4', 'W4']
Y1 --> ['Y2', 'Y3', 'Y5']
W2 --> ['P2', 'R2', 'G2']
Y3 --> ['R3', 'Y1', 'Y5']
Y5 --> ['Y1', 'Y3']
G2 --> ['G5', 'G1', 'W2']
P6 --> ['P4', 'G6']
```

```
          ---------------------------------

W 5 | P 3 | P 4 | W 1 | P 5 | W 3 |

B 3 | Y 4 | G 3 | Y 3 | P 2 | P 6 |

G 5 | G 1 | P 1 | R 5 | G 2 | Y 2 |

B 1 | B 5 | Y 6 | B 6 | G 4 | B 2 |

R 4 | B 4 | W 2 | R 3 | R 6 | R 1 |

G 6 | Y 1 | Y 5 | R 2 | W 6 | W 4 |
```
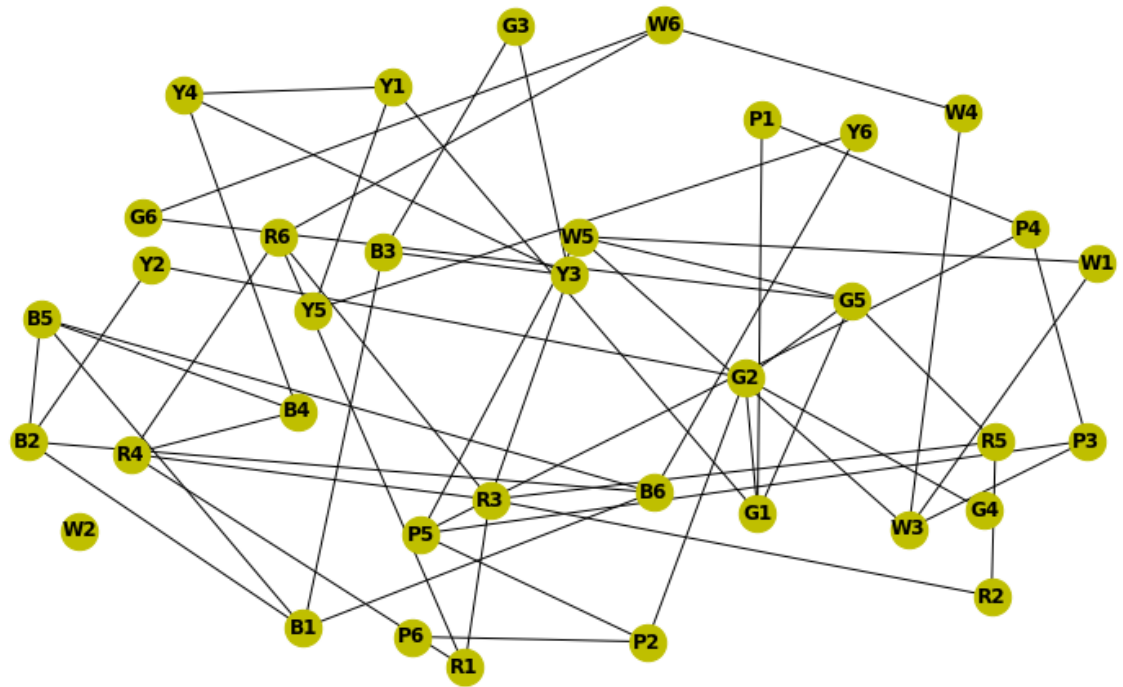
Graph characteristics:
36 nodes
55 edges
Connected? False

```
W5 --> ['G5', 'W1', 'P5', 'W3']
P3 --> ['P4', 'P5', 'W3']
P4 --> ['P3', 'P1', 'P5']
W1 --> ['W5', 'W3']
P5 --> ['W5', 'P3', 'P4', 'P2']
W3 --> ['W5', 'P3', 'W1', 'W4']
B3 --> ['B1', 'G3', 'Y3']
Y4 --> ['B4', 'Y1', 'Y3']
G3 --> ['B3', 'Y3']
Y3 --> ['B3', 'Y4', 'G3', 'R3']
P2 --> ['P5', 'G2', 'P6']
P6 --> ['P2']
G5 --> ['W5', 'G6', 'G1', 'R5', 'G2']
G1 --> ['G5', 'Y1', 'P1', 'G2']
P1 --> ['P4', 'G1']
R5 --> ['G5', 'R3', 'R2']
G2 --> ['P2', 'G5', 'G1', 'G4', 'Y2']
Y2 --> ['G2', 'B2']
B1 --> ['B3', 'B5', 'B6', 'B2']
B5 --> ['B1', 'B4', 'B6', 'B2']
Y6 --> ['Y5', 'B6']
B6 --> ['B1', 'B5', 'Y6', 'B2']
G4 --> ['G2']
B2 --> ['Y2', 'B1', 'B5', 'B6']
R4 --> ['B4', 'R3', 'R6', 'R1']
B4 --> ['Y4', 'B5', 'R4']
W2 --> []
R3 --> ['Y3', 'R5', 'R4', 'R2', 'R6', 'R1']
R6 --> ['R4', 'R3', 'W6', 'R1']
R1 --> ['R4', 'R3', 'R6']
G6 --> ['G5', 'W6']
Y1 --> ['Y4', 'G1', 'Y5']
Y5 --> ['Y6', 'Y1']
R2 --> ['R5', 'R3']
W6 --> ['R6', 'G6', 'W4']
W4 --> ['W3', 'W6']
```

W 2 | P 5 | Y 6 | Y 1 | W 5 | Y 3 |

R 6 | Y 4 | G 6 | R 1 | G 1 | B 5 |

R 3 | Y 5 | P 1 | Y 2 | W 1 | W 4 |

B 1 | B 4 | P 2 | G 2 | P 3 | B 6 |

G 4 | B 2 | W 3 | G 3 | R 4 | R 2 |

G 5 | B 3 | W 6 | R 5 | P 6 | P 4 |

```
Graph characteristics:
36 nodes
48 edges
Connected? True
```

```
----------------------------------

P 1 | Y 3 | R 3 | G 3 | W 6 | P 6 |

W 3 | W 4 | R 2 | Y 6 | B 5 | W 2 |

W 5 | G 5 | R 1 | G 2 | W 1 | R 5 |

R 4 | G 1 | G 4 | P 5 | P 4 | Y 5 |

Y 2 | Y 4 | G 6 | R 6 | B 6 | B 1 |

P 3 | P 2 | Y 1 | B 3 | B 2 | B 4 |
```

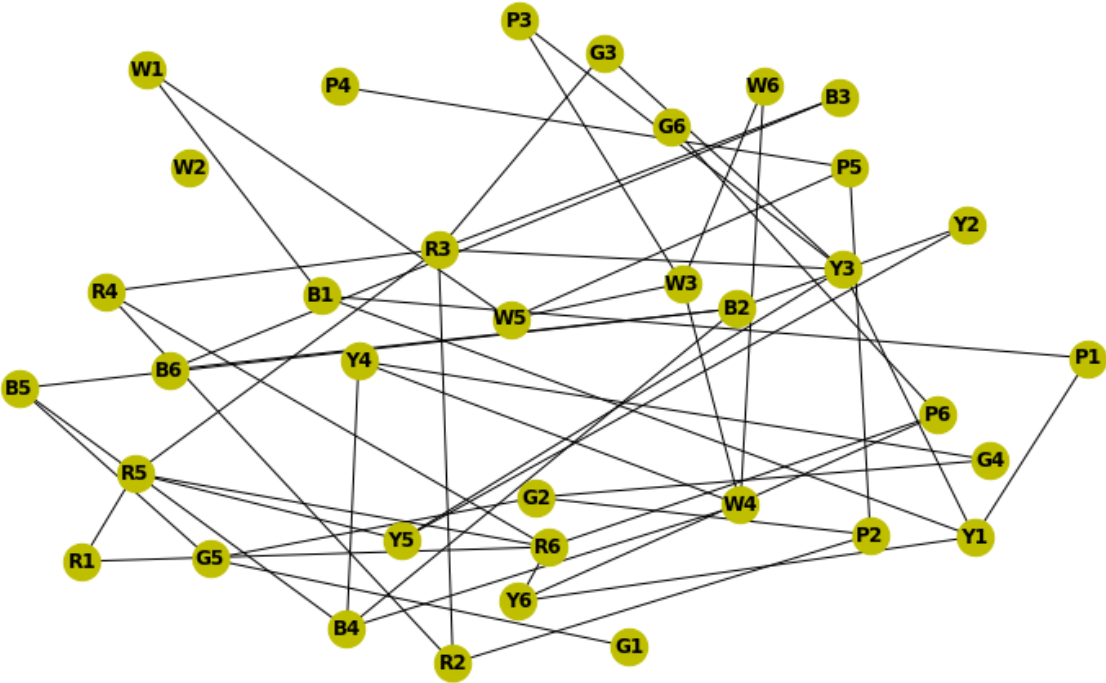Graph characteristics:
36 nodes
54 edges
Connected? True


```
----------------------------------

B 5 | G 2 | G 5 | Y 1 | P 3 | Y 3 |

B 2 | G 4 | B 6 | P 1 | Y 2 | Y 5 |

P 6 | R 1 | W 2 | Y 6 | R 6 | R 5 |

G 6 | P 2 | B 3 | R 2 | R 4 | R 3 |

W 1 | P 5 | P 4 | B 1 | W 5 | G 3 |

B 4 | Y 4 | G 1 | W 4 | W 3 | W 6 |
```

Graph characteristics:
36 nodes
51 edges
Connected? False

```
B5 --> ['B2', 'B4', 'G5']
G2 --> ['G4', 'P2', 'G5']
G5 --> ['B5', 'G2', 'G1']
Y1 --> ['P1', 'Y6', 'B1', 'Y3']
P3 --> ['W3', 'Y3']
Y3 --> ['Y1', 'P3', 'Y5', 'R3', 'G3']
B2 --> ['B5', 'B4', 'B6', 'Y2']
G4 --> ['G2', 'Y4']
B6 --> ['B2', 'B3']
P1 --> ['Y1', 'B1']
Y2 --> ['B2', 'Y5']
Y5 --> ['Y3', 'Y2', 'R5']
P6 --> ['G6', 'Y6', 'R6']
R1 --> ['R6', 'R5']
W2 --> []
Y6 --> ['Y1', 'P6', 'R6']
R6 --> ['P6', 'R1', 'Y6', 'R4', 'R5']
R5 --> ['Y5', 'R1', 'R6', 'R3']
G6 --> ['P6']
P2 --> ['G2', 'P5', 'R2']
B3 --> ['B6', 'R3']
R2 --> ['P2', 'R4', 'R3']
R4 --> ['R6', 'R2', 'R3']
R3 --> ['Y3', 'R5', 'B3', 'R2', 'R4', 'G3']
W1 --> ['B1', 'W5']
P5 --> ['P2', 'P4', 'W5']
P4 --> ['P5']
B1 --> ['Y1', 'P1', 'W1']
W5 --> ['W1', 'P5', 'W3']
G3 --> ['Y3', 'R3']
B4 --> ['B5', 'B2', 'Y4', 'W4']
Y4 --> ['G4', 'B4', 'W4']
G1 --> ['G5']
W4 --> ['B4', 'Y4', 'W3', 'W6']
W3 --> ['P3', 'W5', 'W4', 'W6']
W6 --> ['W4', 'W3']
```

```
          ---------------------------------

Y 4 | P 5 | P 4 | B 5 | Y 1 | P 3 |

Y 3 | G 2 | B 2 | G 4 | Y 6 | R 6 |

W 5 | B 6 | P 1 | Y 5 | R 2 | Y 2 |

R 3 | W 1 | B 3 | W 2 | G 3 | W 4 |

R 4 | P 2 | G 1 | R 5 | B 4 | G 5 |

R 1 | W 6 | P 6 | W 3 | B 1 | G 6 |
```
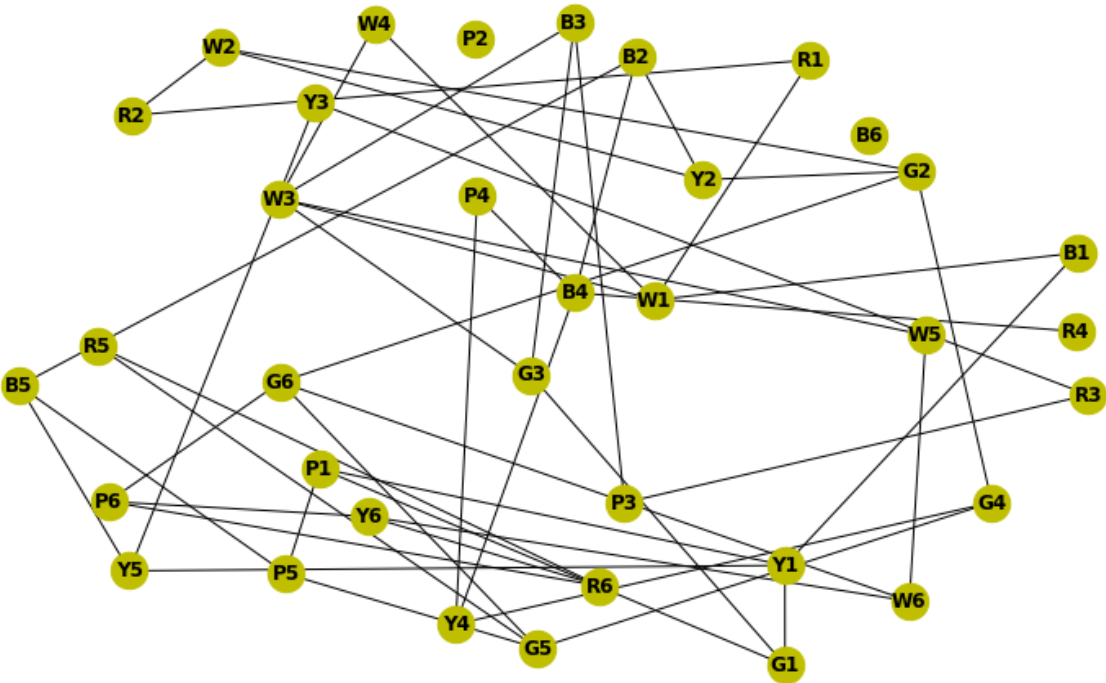
```
Graph characteristics:
36 nodes
52 edges
Connected? True
```

```
----------------------------------
```

B 3 | R 1 | W 4 | W 1 | G 3 | W 3 |

Y 2 | P 5 | B 5 | R 6 | P 1 | B 2 |

G 2 | G 6 | Y 3 | P 6 | R 4 | B 4 |

G 4 | G 5 | P 2 | R 5 | B 6 | Y 4 |

W 2 | R 2 | Y 5 | B 1 | Y 1 | P 4 |

P 3 | W 6 | R 3 | Y 6 | G 1 | W 5 |

Graph characteristics:
36 nodes
49 edges
Connected? False

```
B3 --> ['P3', 'G3', 'W3']
R1 --> ['R2', 'W1']
W4 --> ['W1', 'W3']
W1 --> ['R1', 'W4', 'B1', 'W3']
G3 --> ['B3', 'G1', 'W3']
W3 --> ['B3', 'W4', 'W1', 'G3', 'W5']
Y2 --> ['G2', 'W2', 'B2']
P5 --> ['G5', 'B5', 'P1']
B5 --> ['P5', 'Y5', 'B2']
R6 --> ['P6', 'R5', 'Y6']
P1 --> ['P5', 'Y1', 'G1']
B2 --> ['Y2', 'B5', 'B4']
G2 --> ['Y2', 'G4', 'W2', 'G6']
G6 --> ['G2', 'G5', 'W6', 'P6']
Y3 --> ['Y5', 'R3']
P6 --> ['R6', 'G6', 'Y6']
R4 --> ['B4']
B4 --> ['B2', 'R4', 'Y4', 'P4']
G4 --> ['G2', 'G5', 'Y4']
G5 --> ['P5', 'G6', 'G4', 'R5']
P2 --> []
R5 --> ['R6', 'G5']
B6 --> []
Y4 --> ['B4', 'G4', 'P4']
W2 --> ['Y2', 'G2', 'R2']
R2 --> ['R1', 'W2']
Y5 --> ['B5', 'Y3', 'Y1']
B1 --> ['W1', 'Y1']
Y1 --> ['P1', 'Y5', 'B1', 'G1']
P4 --> ['B4', 'Y4']
P3 --> ['B3', 'R3']
W6 --> ['G6', 'Y6', 'W5']
R3 --> ['Y3', 'P3']
Y6 --> ['R6', 'P6', 'W6']
G1 --> ['G3', 'P1', 'Y1']
W5 --> ['W3', 'W6']
```

```
Y 2 | P 3 | B 2 | R 2 | G 5 | Y 5 |

R 3 | R 4 | R 5 | B 4 | W 6 | Y 6 |

G 3 | Y 4 | W 1 | G 6 | W 4 | W 2 |

P 6 | G 2 | R 1 | Y 1 | P 2 | R 6 |

B 5 | B 6 | B 1 | W 3 | G 1 | P 4 |

Y 3 | W 5 | B 3 | G 4 | P 5 | P 1 |
```
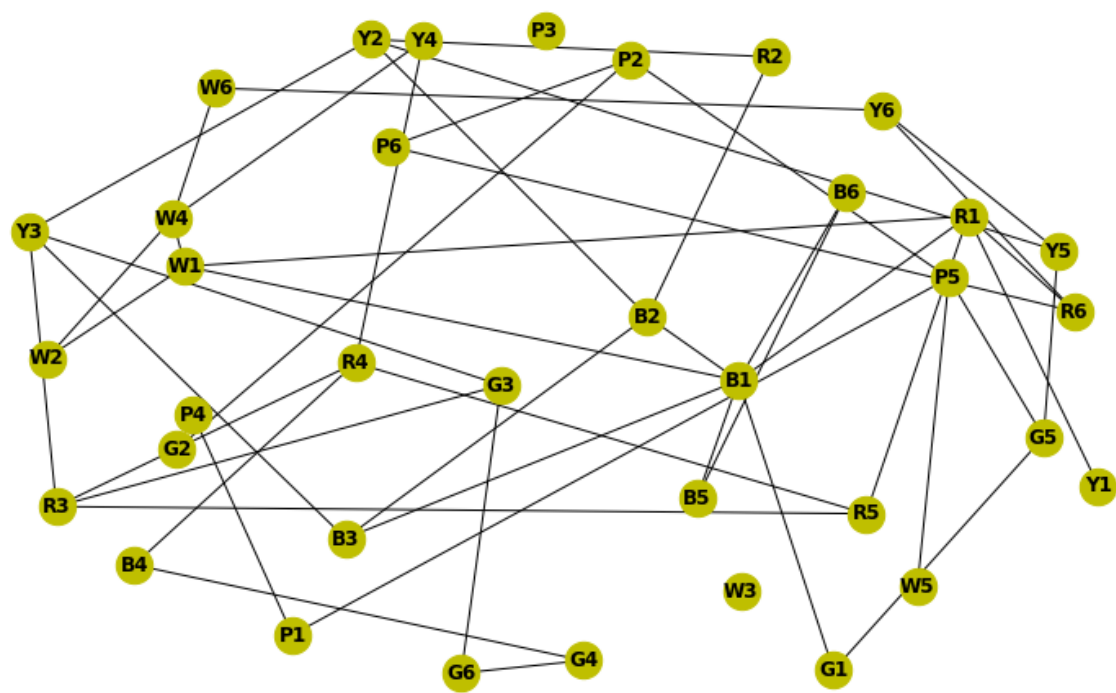
Graph characteristics:
36 nodes
48 edges
Connected? False

MicroRobots

```
Y2 --> ['Y3', 'B2', 'R2', 'Y5']
P3 --> []
B2 --> ['Y2', 'B1', 'B3', 'R2']
R2 --> ['Y2', 'B2']
G5 --> ['G1', 'P5', 'Y5']
Y5 --> ['Y2', 'G5', 'Y6']
R3 --> ['G3', 'Y3', 'R4', 'R5']
R4 --> ['R3', 'Y4', 'R5', 'B4']
R5 --> ['R3', 'R4', 'R1']
B4 --> ['R4', 'G4']
W6 --> ['W4', 'Y6']
Y6 --> ['Y5', 'W6', 'R6']
G3 --> ['R3', 'Y3', 'G6']
Y4 --> ['R4', 'W4']
W1 --> ['R1', 'B1', 'W4', 'W2']
G6 --> ['G3', 'G4']
W4 --> ['W6', 'Y4', 'W1', 'W2']
W2 --> ['W1', 'W4']
P6 --> ['P2', 'R6']
G2 --> ['P2']
R1 --> ['R5', 'W1', 'B1', 'Y1', 'R6']
Y1 --> ['R1']
P2 --> ['P6', 'G2', 'P5']
R6 --> ['Y6', 'P6', 'R1']
B5 --> ['B6', 'B1']
B6 --> ['B5', 'B1']
B1 --> ['B2', 'W1', 'R1', 'B5', 'B6', 'B3', 'G1']
W3 --> []
G1 --> ['G5', 'B1']
P4 --> ['P1']
Y3 --> ['Y2', 'R3', 'G3', 'B3']
W5 --> ['P5']
B3 --> ['B2', 'B1', 'Y3']
G4 --> ['B4', 'G6']
P5 --> ['G5', 'P2', 'W5', 'P1']
P1 --> ['P4', 'P5']
```

```
---------------------------------
```

```
G 1 | P 3 | B 3 | P 2 | Y 5 | B 1 |

G 6 | W 4 | Y 1 | P 5 | G 2 | W 1 |

W 6 | P 6 | P 4 | B 2 | W 3 | Y 3 |

W 2 | G 3 | G 4 | R 1 | B 4 | R 6 |

Y 6 | B 6 | P 1 | R 5 | W 5 | R 3 |

B 5 | G 5 | R 2 | R 4 | Y 4 | Y 2 |
```

```
Graph characteristics:
36 nodes
48 edges
Connected? True
```

```
---------------------------------

Y 3 | G 3 | W 1 | G 2 | W 6 | G 4 |

G 6 | Y 5 | R 2 | Y 4 | R 3 | R 4 |

B 6 | B 5 | G 5 | G 1 | R 6 | B 1 |

P 2 | W 2 | W 4 | P 1 | Y 1 | B 4 |

R 1 | P 4 | W 5 | R 5 | P 6 | Y 2 |

W 3 | Y 6 | P 3 | B 2 | B 3 | P 5 |

Graph characteristics:
36 nodes
50 edges
Connected? False
```
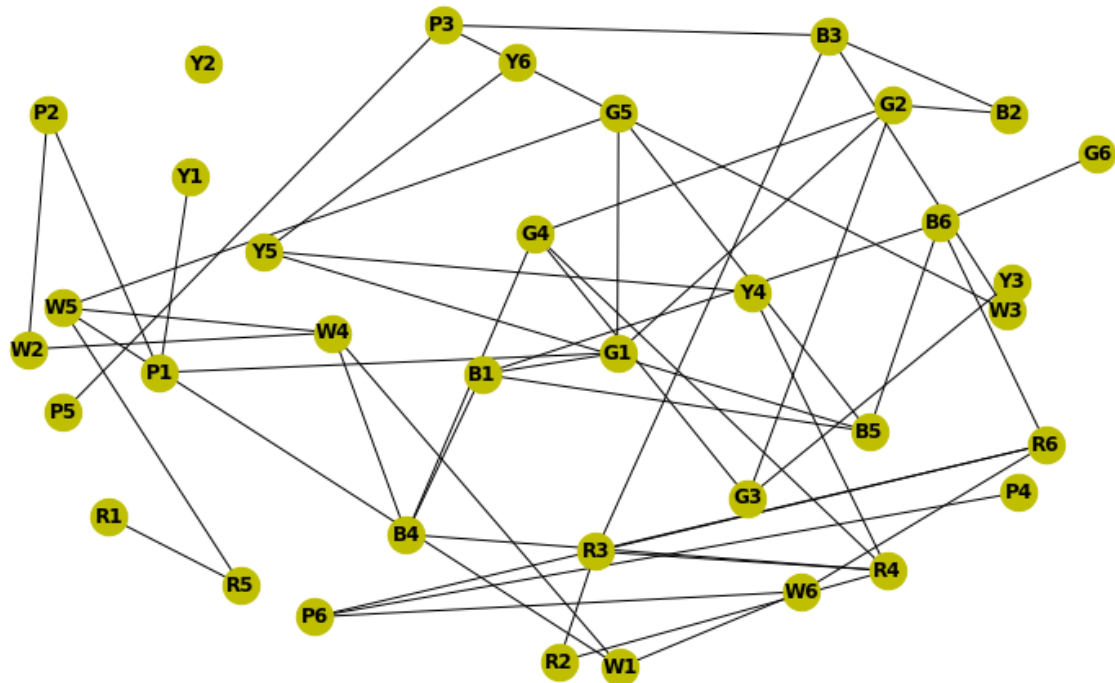

ANSWERING IF THE ACTUAL GAMEBOARD IS ALWAYS CONNECTED

Next, we examine all of the different permutations of the actual board graph and verify they are all connected

```
In [441]:  # implement a function which rotates the boards and places them in the associa
           ted "slot" until all permutations have been achieved.
```

Expected outcome

There are $4!$ ways to place the tiles in the 4 slots. There are 4^4 independent rotations of the individual tiles.

Consider the tile placement without rotation. For each placement, without allowing rotation, there are 3 other equivalent boards. **(add explanation here)**

Consider the tile rotations, each tile has exactly 4 ways of rotating it in place with exactly one other duplicate board created. **(add explanation here)** Therefore, each board is represented by 4 x 4 equivalent arrangements determind by the slot placement and the rotational position of the first tile.

As such, we expect that there are a total of $\frac{4! \times 4^4}{4^2} = 384$ unique boards for four 3×3 tiles.

So expect to see 6144 total boards generated by our program and 5760 of them should be non-unique.

```
In [413]: # implement a function which compares the graphs of two Boards to determine if
          they are, in fact, the same.
```

```
In [ ]:
```