124. Maximum Cut and Bin Packing Problem
Code:

```
import networkx as nx
import itertools

def maximum_cut(graph):
    max_cut_size = 0
    best_partition = None
    nodes = list(graph.nodes())

    for partition in itertools.product([0, 1], repeat=len(nodes)):
        cut_size = 0
        for edge in graph.edges():
            u, v = edge
            if partition[nodes.index(u)] != partition[nodes.index(v)]:
                cut_size += 1

        if cut_size > max_cut_size:
            max_cut_size = cut_size
            best_partition = partition[:]

    return max_cut_size, best_partition

graph = nx.Graph()
graph.add_edges_from([(0, 1), (0, 2), (1, 2), (1, 3), (2, 3)])
max_cut_size, best_partition = maximum_cut(graph)
print("Maximum cut size:", max_cut_size)
print("Best partition:", best_partition)
```

output:

Time complexity:f(n)=o(m*n)