

122. Maximum Cut and Bin Packing Problem

PROGRAM:-

```
def first_fit_decreasing(items, bin_capacity):  
    """  
    First-Fit Decreasing (FFD) algorithm for the Bin Packing problem.  
  
    Parameters:  
    items (list of int): Sizes of the items.  
    bin_capacity (int): Capacity of each bin.  
  
    Returns:  
    list of list of int: List of bins with items.  
    """  
  
    # Sort items in decreasing order  
    items = sorted(items, reverse=True)  
  
    # Initialize an empty list of bins  
    bins = []  
  
    # Place each item in the first bin that can accommodate it  
    for item in items:  
        placed = False  
        for bin in bins:  
            if sum(bin) + item <= bin_capacity:  
                bin.append(item)  
                placed = True  
                break  
        if not placed:  
            bins.append([item])  
  
    return bins  
  
# Example usage:  
items = [4, 8, 1, 4, 2, 1]  
bin_capacity = 10  
  
bins = first_fit_decreasing(items, bin_capacity)  
print("First-Fit Decreasing Bin Packing:", bins)
```

OUTPUT:-

```
First-Fit Decreasing Bin Packing: [[8, 2], [4, 4, 1, 1]]  
  
=== Code Execution Successful ===
```

TIME COMPLEXITY:- $O(V+E)$