

```

# Fractional Knapsack Problem in Python

# Function to calculate the maximum profit for the knapsack
def fractional_knapsack(profit, weights, capacity):
    # Calculate profit-to-weight ratio for each item
    ratio = []
    for i in range(len(profit)):
        ratio.append((i, profit[i] / weights[i]))

    # Sort items based on their ratio in descending order
    ratio.sort(key=lambda x: x[1], reverse=True)

    final_profit = 0 # Maximum profit we can carry
    for i, r in ratio:
        if capacity >= weights[i]:
            # Take the whole item
            final_profit += profit[i]
            capacity -= weights[i]
        else:
            # Take the fraction of the remaining capacity
            final_profit += r * capacity
            break

    return final_profit

if __name__ == "__main__":
    # Example usage
    profit = [60, 100, 120]
    weights = [10, 20, 30]
    capacity = 50

    # Output the final maximum profit that can be carried
    print("Final Profit is:", fractional_knapsack(profit, weights,
    capacity))

```

Final Profit is: 240.0