EXERCISE-111 Knapsack problem using greedy

PROGRAM

```
def knapsack_greedy(weights, values, capacity):
    n = len(weights)
    items = list(zip(weights, values, range(n)))
    items.sort(key=lambda x: x[1] / x[0], reverse=True)
    total_value = 0
    selected_items = []
    for weight, value, index in items:
        if capacity <= 0:
            break
        if weight <= capacity:
            total_value += value
            capacity -= weight
            selected_items.append(index)
    return total_value, selected_items
weights = [10, 20, 30]
values = [60, 100, 120]
capacity = 50
max_value, selected_items = knapsack_greedy(weights, values, capacity)
print("Maximum value:", max_value)
print("Selected items:", selected_items)
```

OUTPUT

```
======= RESTART: C:/Use
Maximum value: 160
Selected items: [0, 1]
```

TIME COMPLEXITY

$O(n\log n)$