

112) Knapsack problem using greedy

CODE:

```
def knapsack_greedy(weights, values, capacity):
    n = len(weights)
    value_per_weight = [(values[i] / weights[i], weights[i], values[i]) for i in
range(n)]
    value_per_weight.sort(reverse=True, key=lambda x: x[0])

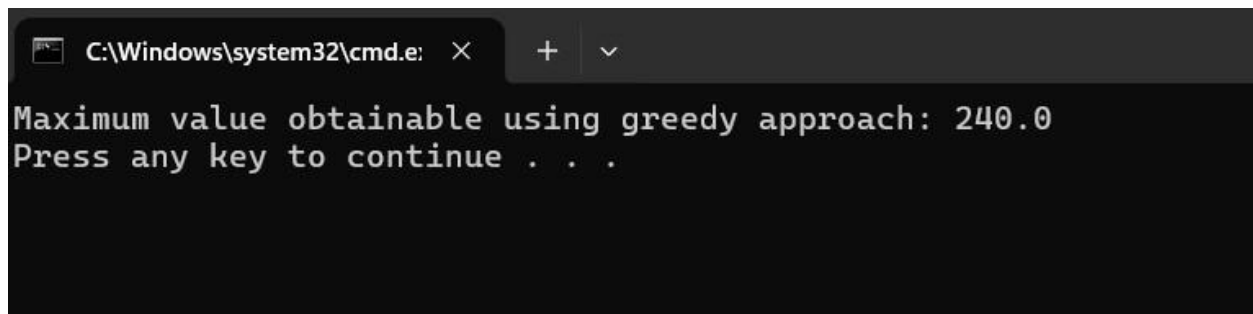
    total_value = 0
    total_weight = 0
    for ratio, weight, value in value_per_weight:
        if total_weight + weight <= capacity:
            total_value += value
            total_weight += weight
        else:
            remaining_capacity = capacity -
total_weight
            total_value += ratio *
remaining_capacity
            break

    return total_value

if __name__ ==
 "__main__":
    weights =
[10, 20, 30]
    values =
[60, 100, 120]
    capacity
= 50

    max_value = knapsack_greedy(weights, values, capacity)
    print(f"Maximum value obtainable using greedy approach: {max_value}")
```

OUTPUT:

A screenshot of a Windows command prompt window. The title bar shows the path 'C:\Windows\system32\cmd.e' and standard window controls. The command prompt displays the output of the program: 'Maximum value obtainable using greedy approach: 240.0' followed by 'Press any key to continue . . .'. The text is in a light green monospace font on a black background.

TIME COMPLEXITY : $O(n)$