```
#include <iostream>
#include <vector>

using namespace std;

struct Item {
    int weight;
    int value;
};

int knapsack(int capacity, vector<Item> items) {
    int n = items.size();

    vector<vector<int>> dp(n + 1, vector<int>(capacity + 1, 0));

    for (int i = 1; i <= n; ++i) {
        for (int w = 0; w <= capacity; ++w) {
            if (items[i - 1].weight > w) {
                dp[i][w] = dp[i - 1][w];
            } else {
                dp[i][w] = max(dp[i - 1][w], items[i - 1].value + dp[i
- 1][w - items[i - 1].weight]);
            }
        }
    }
    return dp[n][capacity];
}

int main() {
    int n, capacity;
    cout << "Enter the number of items: ";
    cin >> n;
```

```
    vector<Item> items(n);

    cout << "Enter the weight and value of each item:" << endl;
    for (int i = 0; i < n; ++i) {
        cin >> items[i].weight >> items[i].value;
    }

    cout << "Enter the maximum capacity of the knapsack: ";
    cin >> capacity;

    int maxValue = knapsack(capacity, items);
    cout << "Maximum value in the knapsack: " << maxValue << endl;

    return 0;
}


//OUTPUT:
Enter the number of items: 3
Enter the weight and value of each item:
10 60
20 100
30 120
Enter the maximum capacity of the knapsack: 50
Maximum value in the knapsack: 220
```