**Fractional Knapsack**

#include <stdio.h>

#define MAX 10

struct Item {

int index;

int value;

int weight;

float ratio;

};

void sortItems(struct Item items[], int n) {

struct Item temp;

for (int i = 0; i < n - 1; i++) {

for (int j = i + 1; j < n; j++) {

if (items[j].ratio > items[i].ratio) {

temp = items[i];

items[i] = items[j];

items[j] = temp;

}

}

}

}

int main() {

struct Item items[MAX];

int n, capacity;

printf("Enter the number of items: ");

scanf("%d", &n);

printf("Enter knapsack capacity: ");

scanf("%d", &capacity);

for (int i = 0; i < n; i++) {

printf("Enter value and weight for item %d: ", i + 1);

scanf("%d %d", &items[i].value, &items[i].weight);

items[i].ratio = (float)items[i].value / items[i].weight;

items[i].index = i + 1;

}

sortItems(items, n);

float totalValue = 0.0;

int remaining = capacity;

printf("\nSelected weights:\n");

for (int i = 0; i < n && remaining > 0; i++) {

if (items[i].weight <= remaining) {

remaining -= items[i].weight;

totalValue += items[i].value;

printf("Item %d: %d (full)\n", items[i].index, items[i].weight);

} else {

float fraction = (float)remaining / items[i].weight;

totalValue += items[i].value \* fraction;

printf("Item %d: %.2f (fractional)\n", items[i].index, (float)remaining);

remaining = 0;

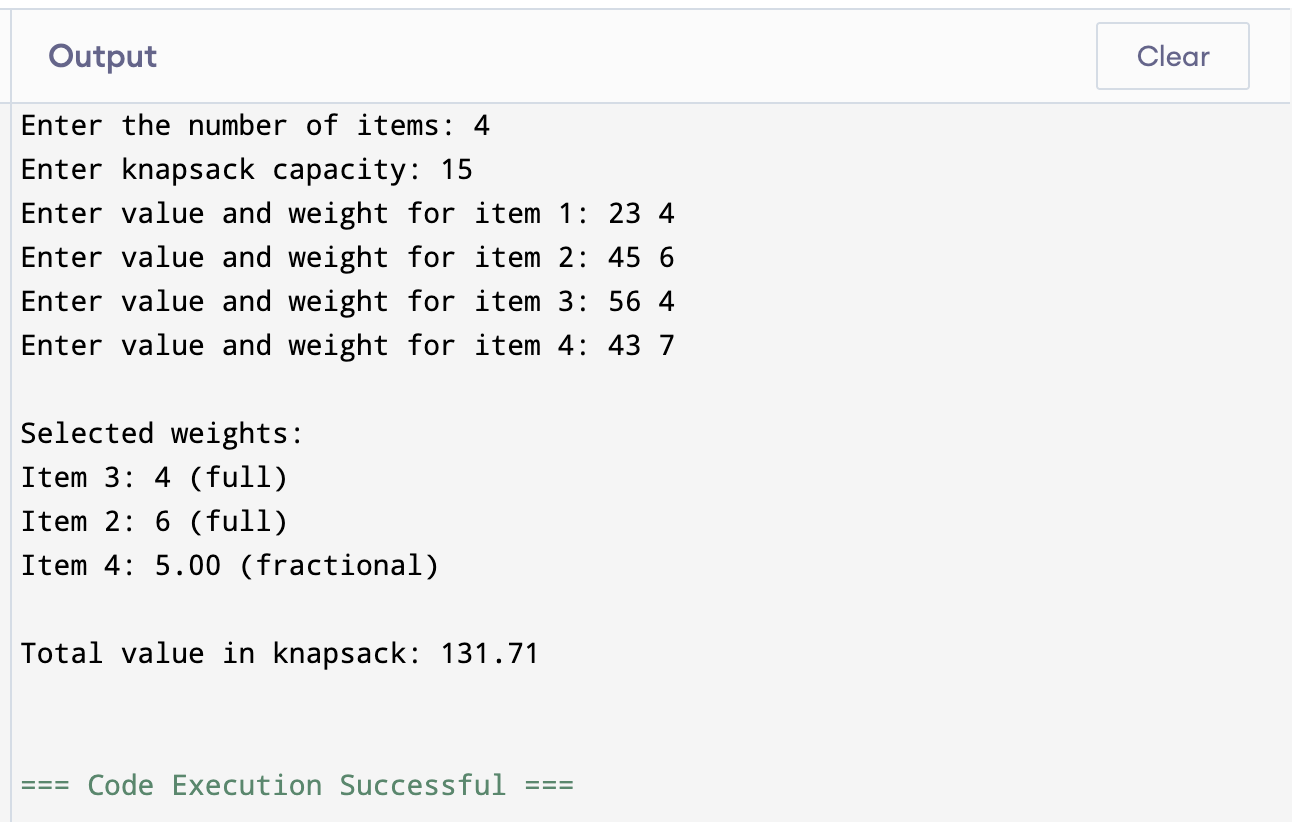
}

}

printf("\nTotal value in knapsack: %.2f\n", totalValue);

return 0;

}

**OUTPUT:**