Softcopy – Practical-3 (Fractical Knapsack)

Code

```
import java.util.*;

public class Code_A3 { // Main public class must match filename
    static class Item {
        int value, weight;
        double ratio;

        Item(int value, int weight) {
            this.value = value;
            this.weight = weight;
            this.ratio = (double) value / weight;
        }
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        List<Item> items = new ArrayList<>();
        int choice;

        while (true) {
            System.out.println("\n=== Fractional Knapsack Menu ===");
            System.out.println("1. Enter items");
            System.out.println("2. Solve fractional knapsack");
            System.out.println("3. Exit");
            System.out.print("Enter your choice: ");
            choice = sc.nextInt();

            switch (choice) {
                case 1:
                    items.clear();
                    System.out.print("Enter number of items: ");
                    int n = sc.nextInt();
                    for (int i = 0; i < n; i++) {
                        System.out.print("Item " + (i + 1) + " value: ");
                        int v = sc.nextInt();
                        System.out.print("Item " + (i + 1) + " weight:
");
                        int w = sc.nextInt();
                        items.add(new Item(v, w));
                    }
                    break;

                case 2:
                    if (items.isEmpty()) {
                        System.out.println("Please enter items first!");
                        break;
                    }
                    System.out.print("Enter knapsack capacity: ");
                    int capacity = sc.nextInt();

                    // Sort by value-to-weight ratio
                    items.sort((a, b) -> Double.compare(b.ratio,
a.ratio));

                    double totalValue = 0;
                    int remaining = capacity;
                    System.out.println("\nItems taken (value/weight
fraction):");
                    for (Item item : items) {
                        if (remaining == 0) break;
                        if (item.weight <= remaining) {
```

```
                            System.out.println(item.value + "/" +
item.weight + " -> Full");
                            totalValue += item.value;
                            remaining -= item.weight;
                        } else {
                            double fraction = (double) remaining /
item.weight;
                            System.out.println(item.value + "/" +
item.weight + " -> " + (fraction * 100) + "%");
                            totalValue += item.value * fraction;
                            remaining = 0;
                        }
                    }

                    System.out.printf("Maximum total value in knapsack:
%.2f\n", totalValue);
                    break;

                case 3:
                    System.out.println("Exiting program. Goodbye!");
                    sc.close();
                    return;

                default:
                    System.out.println("Invalid choice!");
            }
        }
    }
}
```

Output

```
=== Fractional Knapsack Menu ===
1. Enter items
2. Solve fractional knapsack
3. Exit
Enter your choice: 1
Enter number of items: 5
Item 1 value: 3
Item 1 weight: 5
Item 2 value: 6
Item 2 weight: 3
Item 3 value: 3
Item 3 weight: 6
Item 4 value: 3
Item 4 weight: 6
Item 5 value: 2
Item 5 weight: 6

=== Fractional Knapsack Menu ===
1. Enter items
2. Solve fractional knapsack
3. Exit
Enter your choice: 2
Enter knapsack capacity: 7

Items taken (value/weight fraction):
6/3 -> Full
3/5 -> 80.0%
Maximum total value in knapsack: 8.40

=== Fractional Knapsack Menu ===
1. Enter items
2. Solve fractional knapsack
3. Exit
Enter your choice: 3
Exiting program. Goodbye!
```