FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

CSC4202 Design and Analysis of Algorithms

Assignment 2 - Real-Life Carry-On Knapsack

Semester 2 2024/2025

NAME   : CHAN CI EN

MATRIC. NO.   : 215035

LECTURER   : DR. NUR ARZILAWATI BINTI MD YUNUS

Submission Date: 31/5/2025

Results

	Greedy algorithm	Dynamic programming
Total weight	3977g	3995g
Total worth	721	722
Speed	35700ns	4093100ns

```
Comparison Table:
------------------------------------------
                Greedy      Dynamic Programming
Total weight:   3977g          3995g
Total worth:    721            722
Speed (ns):     35700       4093100
------------------------------------------
```

```
Greedy Solution:
Total weight: 3977g
Total worth: 721
Selected items:
- Innergie PocketCell USB-C 6000mAh power bank (Worth: 17, Weight: 14g)
- JBL Reflect Mini Bluetooth Sport Headphones (Worth: 13, Weight: 14g)
- 10 pairs thongs (Worth: 39, Weight: 80g)
- 2 pairs Injinji Women's Run Lightweight No-Show Toe Socks (Worth: 25, Weight: 54g)
- Lightweight Merino Wool Buff (Worth: 20, Weight: 50g)
- Oakley Latch Sunglasses (Worth: 11, Weight: 30g)
- Lenovo X1 Carbon (5th Gen) (Worth: 40, Weight: 112g)
- Laptop Bag (Worth: 7, Weight: 20g)
- 1 fancy tank top (Worth: 24, Weight: 71g)
- Petzl E+LITE Emergency Headlamp (Worth: 8, Weight: 27g)
- 5 pairs black socks (Worth: 26, Weight: 95g)
- 2 pairs Nike Pro shorts (Worth: 30, Weight: 112g)
- Field Notes Pitch Black Memo Book Dot-Graph (Worth: 18, Weight: 68g)
- Peak Design Cuff Camera Wrist Strap (Worth: 6, Weight: 26g)
- Ray Ban Wayfarer Classic (Worth: 10, Weight: 45g)
- 2 Lululemon Cool Racerback (Worth: 36, Weight: 174g)
- Seagate Backup PlusSlim (Worth: 32, Weight: 159g)
- 1 pair Uniqlo leggings (Worth: 37, Weight: 185g)
- The Roost Stand (Worth: 34, Weight: 170g)
- Touchlight (Worth: 2, Weight: 10g)
- 2 Underarmour HeatGear CoolSwitch tank tops (Worth: 27, Weight: 138g)
- 2 pairs Lululemon shorts (Worth: 29, Weight: 184g)
- 1 pair black denim shorts (Worth: 31, Weight: 197g)
- 1 light and stretchylong-sleeve shirt (Gap Fit) (Worth: 23, Weight: 147g)
- Isabella T-Strap Croc sandals (Worth: 28, Weight: 200g)
- Humangear GoBites Duo (Worth: 3, Weight: 22g)
- Anker SoundCore nano Bluetooth Speaker (Worth: 12, Weight: 89g)
- 5 Underarmour Strappy (Worth: 38, Weight: 305g)
- 1 LBD (H&M) (Worth: 19, Weight: 174g)
- Deuter First Aid Kit Active (Worth: 15, Weight: 144g)
- Uniqlo Ultralight Down insulating jacket (Worth: 22, Weight: 235g)
- ThinkPad Compact Bluetooth Keyboard with trackpoint (Worth: 33, Weight: 460g)
- Travelon Micro Scale (Worth: 5, Weight: 125g)
- Vapur Bottle 1L (Worth: 1, Weight: 41g)
```

```
Dynamic Programming Solution:
Total weight: 3995g
Total worth: 722
Selected items:
- Lenovo X1 Carbon (5th Gen) (Worth: 40, Weight: 112g)
- 10 pairs thongs (Worth: 39, Weight: 80g)
- 5 Underarmour Strappy (Worth: 38, Weight: 305g)
- 1 pair Uniqlo leggings (Worth: 37, Weight: 185g)
- 2 Lululemon Cool Racerback (Worth: 36, Weight: 174g)
- The Roost Stand (Worth: 34, Weight: 170g)
- Seagate Backup PlusSlim (Worth: 32, Weight: 159g)
- 1 pair black denim shorts (Worth: 31, Weight: 197g)
- 2 pairs Nike Pro shorts (Worth: 30, Weight: 112g)
- 2 pairs Lululemon shorts (Worth: 29, Weight: 184g)
- Isabella T-Strap Croc sandals (Worth: 28, Weight: 200g)
- 2 Underarmour HeatGear CoolSwitch tank tops (Worth: 27, Weight: 138g)
- 5 pairs black socks (Worth: 26, Weight: 95g)
- 2 pairs Injinji Women's Run Lightweight No-Show Toe Socks (Worth: 25, Weight: 54g)
- 1 fancy tank top (Worth: 24, Weight: 71g)
- 1 light and stretchylong-sleeve shirt (Gap Fit) (Worth: 23, Weight: 147g)
- Uniqlo Ultralight Down insulating jacket (Worth: 22, Weight: 235g)
- Patagonia Torrentshell (Worth: 21, Weight: 301g)
- Lightweight Merino Wool Buff (Worth: 20, Weight: 50g)
- 1 LBD (H&M) (Worth: 19, Weight: 174g)
- Field Notes Pitch Black Memo Book Dot-Graph (Worth: 18, Weight: 68g)
- Innergie PocketCell USB-C 6000mAh power bank (Worth: 17, Weight: 14g)
- Important papers (Worth: 16, Weight: 228g)
- Deuter First Aid Kit Active (Worth: 15, Weight: 144g)
- JBL Reflect Mini Bluetooth Sport Headphones (Worth: 13, Weight: 14g)
- Anker SoundCore nano Bluetooth Speaker (Worth: 12, Weight: 89g)
- Oakley Latch Sunglasses (Worth: 11, Weight: 30g)
- Ray Ban Wayfarer Classic (Worth: 10, Weight: 45g)
- Petzl E+LITE Emergency Headlamp (Worth: 8, Weight: 27g)
- Laptop Bag (Worth: 7, Weight: 20g)
- Peak Design Cuff Camera Wrist Strap (Worth: 6, Weight: 26g)
- Travelon Micro Scale (Worth: 5, Weight: 125g)
- Humangear GoBites Duo (Worth: 3, Weight: 22g)
```

Source Code

```java
1  public class Item {
2      public String name;
3      public int worth;
4      public int weight;
5
6      public Item(String name, int worth, int weight) {
7          this.name = name;
8          this.worth = worth;
9          this.weight = weight;
10     }
11
12     @Override
13     public String toString() {
14         return name + " (Worth: " + worth + ", Weight: " + weight + "g)";
15     }
16 }
```

```java
1  public interface Knapsack {
2      void solve();
3      int getTotalWorth();
4      int getTotalWeight();
5      Item[] getSelectedItems();
6      long getExecutionTime();
7  }
```

```java
public class GreedyKnapsack implements Knapsack {
    private Item[] items;
    private int capacity;
    private boolean[] selected;
    private long executionTime;
    private int totalWorth;
    private int totalWeight;

    public GreedyKnapsack(Item[] items, int capacity) {
        this.items = items.clone();   // Make a copy of the array
        this.capacity = capacity;
        this.selected = new boolean[items.length];
    }

    public void solve() {
        long startTime = System.nanoTime();

        // Sort items by worth/weight ratio using selection sort
        for (int i = 0; i < items.length - 1; i++) {
            int maxIndex = i;
            for (int j = i + 1; j < items.length; j++) {
                if ((double)items[j].worth / items[j].weight >
                    (double)items[maxIndex].worth / items[maxIndex].weight) {
                    maxIndex = j;
                }
            }
            if (maxIndex != i) {
                Item temp = items[i];
                items[i] = items[maxIndex];
                items[maxIndex] = temp;
            }
        }

        // Select items
        totalWeight = 0;
        totalWorth = 0;
        for (int i = 0; i < items.length; i++) {
            if (totalWeight + items[i].weight <= capacity) {
                selected[i] = true;
                totalWeight += items[i].weight;
                totalWorth += items[i].worth;
            }
        }

        executionTime = System.nanoTime() - startTime;
    }

    public int getTotalWorth() {
        return totalWorth;
    }

    public int getTotalWeight() {
        return totalWeight;
    }

    public Item[] getSelectedItems() {
        int count = 0;
        for (boolean s : selected) {
            if (s) count++;
        }

        Item[] selectedItems = new Item[count];
        int index = 0;
        for (int i = 0; i < items.length; i++) {
            if (selected[i]) {
                selectedItems[index++] = items[i];
            }
        }
        return selectedItems;
    }

    public long getExecutionTime() {
        return executionTime;
    }
}
```

```java
import java.util.Arrays;

public class DPKnapsack implements Knapsack {
    private Item[] items;
    private int capacity;
    private int[][] dp;
    private boolean[] selected;
    private long executionTime;

    public DPKnapsack(Item[] items, int capacity) {
        this.items = items.clone();  // Make a copy of the array
        this.capacity = capacity;
        this.dp = new int[items.length + 1][capacity + 1];
        this.selected = new boolean[items.length];
    }

    public void solve() {
        long startTime = System.nanoTime();

        // Initialize first row with zeros
        for (int j = 0; j <= capacity; j++) {
            dp[0][j] = 0;
        }

        // Fill the dp table
        for (int i = 1; i <= items.length; i++) {
            Item item = items[i-1];
            for (int j = 0; j <= capacity; j++) {
                // By default, inherit previous state
                dp[i][j] = dp[i-1][j];

                // Try to include current item if it fits
                if (j >= item.weight) {
                    int withItem = item.worth + dp[i-1][j - item.weight];
                    if (withItem > dp[i][j]) {
                        dp[i][j] = withItem;
                    }
                }
            }
        }

        // Trace back
        Arrays.fill(selected, false);
        int j = capacity;
        int remainingCapacity = capacity;
        int remainingProfit = dp[items.length][j];

        for (int i = items.length; i > 0 && remainingCapacity > 0 && remainingProfit > 0; i--) {
            Item item = items[i-1];
            // Check if this item was included by comparing with previous state
            if (remainingCapacity >= item.weight && dp[i][remainingCapacity] != dp[i-1][remainingCapacity]) {
                selected[i-1] = true;
                remainingProfit -= item.worth;
                remainingCapacity -= item.weight;
            }
        }

        executionTime = System.nanoTime() - startTime;
    }

    public int getTotalWorth() {
        return dp[items.length][capacity];
    }

    public int getTotalWeight() {
        int weight = 0;
        for (int i = 0; i < items.length; i++) {
            if (selected[i]) {
                weight += items[i].weight;
            }
        }
        return weight;
    }

    public Item[] getSelectedItems() {
        int count = 0;
        for (boolean s : selected) {
            if (s) count++;
        }

        Item[] selectedItems = new Item[count];
        int index = 0;
        for (int i = 0; i < items.length; i++) {
            if (selected[i]) {
                selectedItems[index++] = items[i];
            }
        }
        return selectedItems;
    }

    public long getExecutionTime() {
        return executionTime;
    }
}
```

```java
public class Main {
    public static void main(String[] args) {
        Item[] items = new Item[] {
            new Item("Lenovo X1 Carbon (5th Gen)", 40, 112),
            new Item("10 pairs thongs", 39, 80),
            new Item("5 Underarmour Strappy", 38, 305),
            new Item("1 pair Uniqlo leggings", 37, 185),
            new Item("2 Lululemon Cool Racerback", 36, 174),
            new Item("Chargers and cables in Mini Bomber Travel Kit", 35, 665),
            new Item("The Roost Stand", 34, 170),
            new Item("ThinkPad Compact Bluetooth Keyboard with trackpoint", 33, 460),
            new Item("Seagate Backup PlusSlim", 32, 159),
            new Item("1 pair black denim shorts", 31, 197),
            new Item("2 pairs Nike Pro shorts", 30, 112),
            new Item("2 pairs Lululemon shorts", 29, 184),
            new Item("Isabella T-Strap Croc sandals", 28, 200),
            new Item("2 Underarmour HeatGear CoolSwitch tank tops", 27, 138),
            new Item("5 pairs black socks", 26, 95),
            new Item("2 pairs Injinji Women's Run Lightweight No-Show Toe Socks", 25, 54),
            new Item("1 fancy tank top", 24, 71),
            new Item("1 light and stretchylong-sleeve shirt (Gap Fit)", 23, 147),
            new Item("Uniqlo Ultralight Down insulating jacket", 22, 235),
            new Item("Patagonia Torrentshell", 21, 301),
            new Item("Lightweight Merino Wool Buff", 20, 50),
            new Item("1 LBD (H&M)", 19, 174),
            new Item("Field Notes Pitch Black Memo Book Dot-Graph", 18, 68),
            new Item("Innergie PocketCell USB-C 6000mAh power bank", 17, 14),
            new Item("Important papers", 16, 228),
            new Item("Deuter First Aid Kit Active", 15, 144),
            new Item("Stanley Classic Vacuum Camp Mug 16oz", 14, 454),
            new Item("JBL Reflect Mini Bluetooth Sport Headphones", 13, 14),
            new Item("Anker SoundCore nano Bluetooth Speaker", 12, 89),
            new Item("Oakley Latch Sunglasses", 11, 30),
            new Item("Ray Ban Wayfarer Classic", 10, 45),
            new Item("Zip bag of toiletries", 9, 236),
            new Item("Petzl E+LITE Emergency Headlamp", 8, 27),
            new Item("Laptop Bag", 7, 20),
            new Item("Peak Design Cuff Camera Wrist Strap", 6, 26),
            new Item("Travelon Micro Scale", 5, 125),
            new Item("BlitzWolf Bluetooth Tripod/Monopod", 4, 150),
            new Item("Humangear GoBites Duo", 3, 22),
            new Item("Touchlight", 2, 10),
            new Item("Vapur Bottle 1L", 1, 41)
        };

        int capacity = 4000;

        System.out.println("Dynamic Programming Solution:");
        DPKnapsack dp = new DPKnapsack(items, capacity);
        dp.solve();
        printResults(dp);

        System.out.println("\nGreedy Solution:");
        GreedyKnapsack greedy = new GreedyKnapsack(items, capacity);
        greedy.solve();
        printResults(greedy);

        System.out.println("\nComparison Table:");
        System.out.println("-------------------------------------");
        System.out.println("                Greedy    Dynamic Programming");
        System.out.println("Total weight:      " + greedy.getTotalWeight() + "g      " + dp.getTotalWeight() + "g");
        System.out.println("Total worth:       " + greedy.getTotalWorth() + "        " + dp.getTotalWorth());
        System.out.println("Speed (ns):        " + greedy.getExecutionTime() + "     " + dp.getExecutionTime());
        System.out.println("-------------------------------------");
    }

    private static void printResults(Knapsack knapsack) {
        System.out.println("Total weight: " + knapsack.getTotalWeight() + "g");
        System.out.println("Total worth: " + knapsack.getTotalWorth());
        System.out.println("Selected items:");
        for (Item item : knapsack.getSelectedItems()) {
            System.out.println("- " + item);
        }
    }
}
```