Crash Course Example Questions

Yuetong Guo

**1. (Fill-in-the-blank)**

The 0/1 Knapsack problem is named 0/1 because each item can either be taken \_\_\_\_\_ or not taken \_\_\_\_\_.

**Answer:** entirely (1), at all (0)

**2. (Multiple Choice)**

The time complexity of the dynamic programming solution for the 0/1 Knapsack problem is:

a) O(n)

b) O(n^2)

c) O(nW)

d) O(2^n)

**Answer:** C

**3. (Multiple Choice)**

For the following 0/1 Knapsack problem instances, choose the optimal set of items to maximize the total value without exceeding the capacity of the knapsack.

Knapsack capacity: 10

Items: {(5, 3), (8, 4), (9, 5)}

a) {5, 8}

b) {5, 9}

c) {8, 9}

d) {5, 8, 9}

**Answer:** B

**4. (Multiple Choice)**

The 0/1 Knapsack problem is a classic example of problems that can be solved using which kind of optimization technique?

a) Local search

b) Unconstrained optimization

c) Combinatorial optimization

d) Convex optimization

**Answer:** C

**5. (Fill-in-the-blank)**

When using a dynamic programming approach to solve the 0/1 Knapsack problem, a common choice for the state is the tuple (i, W), where i is the index of the current item and W is the \_\_\_\_\_.

**Answer:** remaining weight

**6. (Multiple Choice)**

Which of the following is NOT a characteristic of the 0/1 Knapsack problem?

a) Each item has a weight and a value.

b) An item can be taken partially.

c) The knapsack has a weight limit.

d) The goal is to maximize the total value.

**Answer:** B

**7. (Multiple Choice)**

The 0/1 Knapsack problem can be formulated as an Integer Linear Programming (ILP) problem. Which of the following is a constraint in this formulation?

a) Sum of item weights multiplied by the corresponding decision variables <= knapsack capacity

b) Sum of item values multiplied by the corresponding decision variables >= knapsack capacity

c) Decision variables are real numbers between 0 and 1

d) Decision variables are real numbers between -1 and 1

**Answer:** A

**8. (multiple choice)**

For the following 0/1 Knapsack problem instances, choose the optimal set of items to maximize the total value without exceeding the capacity of the knapsack.

Knapsack capacity: 20

Items: {(3, 5), (5, 8), (8, 11), (10, 14)}

a) {3, 5, 8}

b) {3, 5, 10}

c) {5, 8, 10}

d) {3, 8, 10}

**Answer:** A

**9. (Fill-in-the-blank)**

In a recursive implementation of the dynamic programming solution for the 0/1 Knapsack problem, memorization is used to avoid \_\_\_\_\_.

**Answer:** redundant computations / overlapping subproblems

**10. (Short Answer)**

Can the greedy approach always find the optimal solution for the 0/1 Knapsack problem? Why or why not?

**Answer:** No, the greedy approach cannot always find the optimal solution for the 0/1 Knapsack problem because choosing the item with the best value-to-weight ratio at each step might not lead to the best overall solution.