Analysis and Design of Algorithms

Algorithms

CS3230

Tutorial

Week 12

Question 1

In the graph coloring problem, you are given an undirected graph G = (V,E) and asked to color the vertex with a set of colors such that the colors of v_1 and v_2 must be different if $(v_1,v_2) \in E$. In the optimization version of the problem, the task is to find the minimum number of colors required to color the graph. In the decision version, the question is whether there is a coloring of the graph using *k* or fewer colors. Select all true statements.

1. If we can solve the optimization problem for graph coloring in polynomial time, we would be able to solve the decision problem for graph coloring in polynomial time.
2. If we can solve the decision problem for graph coloring in polynomial time, we would be able to solve the optimization problem for graph coloring in polynomial time.
3. If the decision problem for graph coloring cannot be solved in polynomial time, the optimization problem for graph coloring cannot be solved in polynomial time.
4. If the optimization problem for graph coloring cannot be solved in polynomial time, the decision problem for graph coloring cannot be solved in polynomial time.

Question 2

Consider the two problems:

PARTITION: Given a set of positive integers S, can the set be partitioned into two sets of equal total sum?

BALL-PARTITION: Given k balls, can we divide the balls into two boxes with an equal number of balls?

We try to show that PARTITION $\leq_P$ BALL-PARTITION using the following transformation A:

1) From the problem PARTITION, we are given S, a set of positive integers.

2) We define k as the total sum of all the elements in S.

3) We use this number k for the BALL-PARTITION problem.

What is wrong with this transformation?

- The transformation is correct.
- A YES solution to $A(S)$ does not imply a YES solution to S.
- A YES solution to S does not imply a YES solution to $A(S)$
- The transformation does not run in polynomial time.

Question 3

PARTITION problem: Given a set T of nonnegative integers, can we partition T into two sets of equal total sum?

KNAPSACK problem: Given n items described by non-negative integer pairs (w_1,v_1), ... (w_n,v_n), capacity W and threshold V, is there a subset of item with total weight at most W and total value at least V?

PARTITION instances with total weights that are odd cannot be partitioned into two equal weight sets, hence can immediately be answered with a NO answer. Consider the following transformation of PARTITION instances with total weights that are even numbers into instances of KNAPSACK:

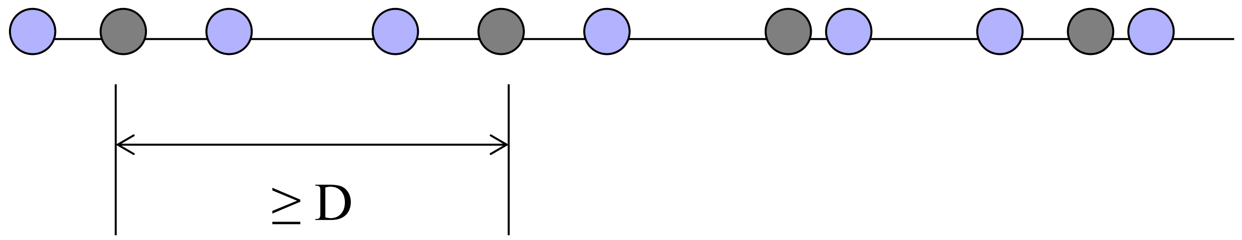
Given a PARTITION instance $\{w_1, .., w_n\}$ with total sum $S = \sum_{i=1}^{n} w_i$, construct a KNAPSACK instance (w_1,w_1), ... (w_n,w_n) with capacity $W = S/2$ and threshold $V = S/2$.

Select all true statements.

1. A YES answer to the PARTITION instance implies a YES answer to the KNAPSACK instance.
2. A YES answer to the KNAPSACK instance implies a YES answer to the PARTITION instance
3. The transformation runs in polynomial time

Question 4

YuckDonald wants to open as many of its chain restaurant on Orchard Road as possible. It has found n suitable locations for its restaurants $a_1, \ldots, a_n$. The restaurants should be at least D distance apart so that they do not compete with each other.



Question 4(a)

Describe how the problem can be modeled as a maximum independent set problem, where an independent set in a graph $G = (V, E)$ is a subset of vertices V such that no two vertices in the graph is connected by an edge.

Question 4(b)

Look for optimal substructure in YuckDonalds problem and argue that

$$M(V) = 1 + M(V - \{a_j \in V : d(a_i, a_j) < D\})$$

where $M(V)$ is the size of the largest set of restaurants that can fit in V, a_i is an element in an optimal solution and $d(a_i, a_j)$ is the distance between a_i and a_j.

Question 4(c)

NUS
National University of Singapore

Assume that the locations for a set of restaurant $a_1, \ldots, a_n$ is sorted and lie on a straight line. Show that a_1 is part of some optimal solution, i.e. selecting the smallest element satisfies the greedy choice property.

Question 4(d)

Complete your greedy algorithm for evaluating $M(V)$ and argue its correctness by mathematical induction.

Question 4(e)

- Since we have a polynomial time algorithm for the problem, we also get a polynomial time algorithm for maximum independent set problem.
  True or False?