Algorithm Design and Analysis: Homework 4

Due on Nov 18, 2017 at 10:15am

14784547 luochenqi

Luochenqi

Problem 1

Solutions

1.Firstly,we want to check if the problem is NP problem, we only need to find of the conjunctive normal form formula is tree,we can judge that in ploynominal time.so problem $\in$ NP.
2.We assume k=n,and from 3satproblem we construct instance: $x \cup y \cup z$. in that case,it will be satisfied by assignment in whitch 3 vairable are given.
3. Obviously ,it is the same.We can deduce 3SAT from this instance and we can deduce this instance from 3-Sat

Problem 2

You are given a directed graph G = (V,E) with weights we on its edges e E. The weights can be negative or positive. The Zero-Weight-Cycle Problem is to decide if there is a simple cycle in G so that the sum of the edge weights on this cycle is exactly 0. Prove that this problem is NP-complete.

Solutions

1.Firstly,given a simple cycle in G,we can determine whether the weights sum of the edges is zero in polynominal time.So Thus Zero-Cycle-Weight-Cycle $\in$ NP.
2.We can reduce this problem to Hamilton cycle problem, building a graph G_1 as follows:
1)$w(e) = -(|V| - 1)$
2)$w(e') = 1$
run ZWC on every G'_e if there is at least one return true ,then return true for hamiltonian-cycle. 3.(a) if a run of ZWC returns True,then we have a cycle of weight zero in some G'_e it must include the only negative edge e with $w(e) = -(|V| - 1)$.as all the other edge w(e')=1,the cycle must also contain $|V| - 1$ edges,indication in all that is a hamiltonian cycle.
(b)if run of hamilonian cycle is true it means visits every node exactly once ,the cycle must have V edges,the cycle contains $w(e) = -(|V| - 1)$ and $w(e') = 1$ so the weight of cycle is zero.
3.if we find a hamilton-cycle with path addition to 0 ,we find the zeroweight-cycle.If we find the zero-weight-cycle ,we can know this graph have the

Problem 3

Solutions

1.Firstly,we can determine if we can find a subset of K nodes in polynominal time.
2.We can reduce this problem is that vertex cover problem to subset problem that we can determine if we have a vertex cover S with size of K.we arbitrarily choose an edge e in G and it must have one node, it must have one node in S,and its two nodes must exsit at least one node in V-S,if each two nodes in V-s doesn't have an edge ,V-S is an independent subset.

3.Suppose we have k nodes coverA,each A will fit the element in C,making to C_1,since A is vertex cover,it covers all the edges,so $\cup c = S$,and $|C_1| \leq k$
Suppose C has C_1 and $\cup c = S$,and $|C_1| \leq k$,C_1 elements making set A,can cover every edge in G,and due to $|C_1| \leq k$,A is G at least node cover.

Problem 4

1.Firstly,for each courses, if student have more than one course in the same slot,exsits a collsion,it can be found in polynominal time.
2.We can reduce this problem to 3-color problem.given any 3 color instance with 3 vertex and 2 edges.Scheduling instance like this.

$$K = 0; C = a, b, c; S = 1, 2; R = \{\{a, b\}, \{b, c\}\}$$

Then we can deduce the 3-color to n color problem.
3.in n color problem the number of same color in adjacent nodes equals to the number of collsion in our problem.Since three-color problem is npc, our problem is NP-complete.

Problem 5

1.We can check if exsits two cycle in polynominal time.
2.We can do the reduction from hamiltonian Cycle.To be sure the distances will become non-negative integers,we use a reduction from hamiltonian cycle so that we define the distances ourselves,depending on the presence of an edge.we can make other driver busy by including additional vertex with distance $\frac{k}{2}$ from s and distance $K + 1$ to all other addresses.
3.from Graph,we make distance $d(v, w) = 2$ if there is an edge in the graph,$d(v, w) = 4$if there is not.and label on of these vertex be the s.add vertex 0 let $d(s, 0) = d(0, s) = n$ and $d(v, 0) = d(0, v) = 2n + 1$for $v \neq s$. K is equal to 2n.

Problem 6

Consider the Knapsack problem. We have n items, each with weight aj and value cj (j = 1, ..., n). All aj and cj are positive integers. The question is to find a subset of the items with total weight at most b such that the corresponding profit is at least k (b and k are also integers). Show that Knapsack is NP-complete by a reduction from Subset Sum.
Solutions
1.Given an input set,it is easy to check if the total weight is at most b and if the total value is at most k,it takes linear times to add the weight and the value of all goods in subset to find if the result is true or false.
2.We can reduce the problem to subset problem with $a_i = c_i = S; b = K = t$,the subset problem is
$\sum a_i \leq b \iff \sum S_i \leq t$
$\sum c_i \geq K \iff \sum Si \geq t$
so $\sum S_i = t$
3.we have a yes answer to the new problem,it means we can find such a subset that satisfied the left part and right part then the subset is a solution to the problem .