Assignment 7 of Algorithm Design and Analysis

L.J.SHOU 寿林钧
No. **201228013229133**

Jan.02 2013

1 NP-Completeness

The subgraph-isomorphism problem takes two graphs G_1 andG_2 and asks whether G_1 is isomorphic to a subgraph ofG_2 . Show that the subgragh-isomorphism problem is NP-complete.
解：
First we prove that the Subgraph Isomorphism problem is in NP. The certificate is $(G_1 = (V_1, E_1), G_2 = (V_2, E_2), \phi V_1 \to V_2)$. The verifying algorithm checks if ϕ is a one- to-one function, and for all u, v $\in$ V1 whether (u, v)$\in E_1$ if and only if $(\phi(u), \phi(v)) \in E_2$.
Secondly, we prove that CLIQUE$\leq_P$ Subgragh Isomorphism. Let$(G = (V, E), k)$ be an input instance for CLIQUE. DefineG_1 to be the complete graph on k vertices, andG_2 to be the grapgh G. Then $(G_1, G_2) \in$ Subgragh Isomorphism if and only if$(G, k) \in$ CLIQUE.

3 NP-Completeness

The set-partition problem takes as input a set S of numbers. The question is whether the numbers can be partitioned into two sets$A$ and B and $B = S`A$ such that $\sum_{x \in A} x = \sum_{x \in B} x$. Show that the set-partition problem is NP-complete.

Proof:
First we show that Set-partition is in NP. The certificate for the Set-partition problem consists of the two sublists S_1 and S_2. Given the sublists, in polynomial time we can compute the sums of the elements in these lists and verify that they are equal. Subset Sum $\leq_p$ set-partition.
Goal:

Next we want to show that SubsetSum(SS) is polynomially reducible to the set-partition problem (PART). That is, we want a polynomial time computable function f, which gives an instance of SS (a set of numbers $S = x_1, x_2, ..., x_n$ and a target value t) outputs an instance of PART (a set of numbers $S2 = x'_1, x'_2, ..., x'_n$) such that S has a subset summing to t if and only if S' can be partitioned into subsets S1 and S2 that sum to the same value.

Transformation:
Observe that the Set-partition problem is a special case of the subset sum problem where we are trying to find a set of numbers S1 that sum to half the total sum of the whole set. Let T be the sum of all the numbers in S. $T = sum_{i=1}^{n} x_i$ If $t = T/2$ then the subset sum problem is an instance of the partition problem, and we are done. If not, then the reduction will create a new number, which if added to any subset that sums to t, will now cause that set to sum to half the elements of the total. The problem is that when we add this new element, we change the total as well, so this must be done carefully. We may assume that $t \leq T/2$, since otherwise the subset sum problem is equivalent to searching for a subset of size$T - t$, and then taking the complement. Create a new element $x_0 = T - 2t$, and call partition on this modified set. Let S' be S together with this new element: $S' = S \cup x$. Clearly the transformation can be done in polynomial time.
Correctness: To see why this works, observe that the sum of elements in S ′ is $T + T - 2t = 2(T - t)$. If there is a solution to the subset sum problem, then by adding in the element x_0 we get a collection of elements that sums to $t + (T - 2t) = T - t$, but this is one half of the total $2(T - t)$, and hence is a solution to the Set-partition problem. Conversely, if there is a solution to this Set-partition problem, then one of the halves of the partition contains the element x0, and the remaining elements in this half of the partition must sum to $(T - t) - (T - 2t) = t$. Thus these elements (without x_0) form a solution to the subset sum problem.