<Discrete Mathematics>

작성자_2018320161_송대선

작성일_06_10

1. 어떻게 k clique 문제의 Search version이 Decision version으로 reduce 되는가
2. 어떻게 3 cnf sat 문제가 Halting problem 으로 reduce 되는가
3. Euclidean algorithm
4. bipartite 그래프의 정의
5. matching 의 정의

(1) to show this reduction, we start with a decider for the decision problem - let's call the decider, D

consider an example graph that consists of 4 nodes, a, b, c, d and 4 edges exist - (a,c), (c,d), (a,d), (b,d)

let's call this graph G and assume that we want to solve the Search version with input (G, 3)

all that we can use here is the decider D that can correctly answers "yes".

now, since there is a clique of size 3, we need to identify all 3 nodes which form a clique - how can we solve this? well, using the following steps, it is guaranteed that the nodes can be found.

- run D with input {a,b,c}, 3
- run D with input {a,c,d}, 3
- run D with input {a,b,d}. 3
- run D with input {b,c,d}, 3

$_4C_3 = 4$

because "there are 3 nodes that form a clique" and "D is assume to work correctly", it is clear that one of these must return yes - in this example, the second says "yes" and we have found an answer! of course, if we are given a different example, the same idea will be applied again and again.

D 가 이미 yes 라고 답을 했으므로 노드 4 개 중 어떤 3 개의 경우
D 는 분명 yes 를 출력합니다. 따라서 모든 가능성들을 체크해서
그 중 답이 yes 가 되게 하는 노드들이 clique 을 만들게 되고
이 아이디어를 이용하여 다른 예제들에 대해서도 설명할 수 있습니다.

(2) 정지문제를 해결하는 알고리즘이 HALT(P,x) 라고 한다면 우리는
다음과 같은 프로그램 A 를 만들어 A 와 A 의 입력을 HALT 에 넣어서
해결할 수 있습니다.

program A (f)

만일 f 가 satisfiable 하면 정지, 그렇지 않으면 무한 루프

여기서 f 가 satifiable 한지 아닌지는 어차피 유한개의 boolean 변수들에
모든 가능한 값들 대입해 보면 알 수 있으므로 유한 단계 안에 체크가
가능하고

A 와 f 를 HALT 의 입력으로 넣어 나온 결과가 yes 즉 halt 한다면 f 는
satifiable, 그렇지 않다면 unsatifiable 하다고 결론 내릴 수 있습니다.

(3) Euclidean algorithm (refer to the algorithm on page 224 of the textbook)

input: A, B //A > B >= 0

a:= A;
b:= B;
r:= B;

while (b is not equal to 0)

r:= a mod b   // mod is an operator which returns the remainder when a is divided by b
a:= b:
b:= r;

return a   //a is the gcd of A and B

this algorithm is based on the following property that appears on page 221 of
the textbook

if a and b are integers not both zero AND
if q and r are positive integers that satisfy a = b * q + r
then gcd(a,b) = gcd(b,r)

for example, gcd(120,50) = gcd(50, 20) because
120 = 50 * 2 + 20 in which a and b are 120 and 50, and b and r are 50 and 20,

respectively. now this can be repeated as follows:
gcd(50,20) = gcd(20, 10), again
gcd(20,10) = gcd(10,0) = 10

the significance if this property lies in that after a finite number of steps, it is guaranteed to stop with the gcd of two numbers and b.

(4) bipartite 그래프는 unidrected 그래프이면서 그 그래프의 노드들의 집합인 V 를 정확하게 두개의 집합들 A 와 B 로 나눌 수 있되 다음 조건을 만족시키는 것을 말합니다.

- A 와 B 모두 공집합이 아니고
- A 와 B 의 교집합이 공집합이고
- A 와 B 모두 V 의 부분집합들이고
- 모든 edge 들이 A 와 B 사이에만 존재 (즉 하나의 edge 라고 같은 집합 안에 존재하지 않음)

(5) undirecte 그래프 G=(V,E) 가 주어질 때 이 그래프 안의 matching M 은 E 의 부분집합이면서 다음 조건을 만족하는 것입니다.

- V 의 모든 원소 x 에 대해
- M 안에 많아야 한개 혹은 0 개의 edge y 가 존재하며
- y is incident on x

즉, 이 말은 이 그래프 안의 모든 노드들 하나 하나에 대해
그 노드를 touch 하는 edge 가 많아야 한개 혹은 0 개 M 에 존재한다는 말입니다.