Algorithmn HW10

5140379032 JIN YI FAN

Problem 10.3

Require: a given undirected graph $G = (V, E)$
Ensure: whether it can be 2-colored

```
function DFS(v)
    mark v visited
    v.mark ← (color ←!color)        ▷ mark neighbour vertexes with different colors
    for each edge (v, w) ∈ E do
        if w is marked unvisited then
            DFS(w)
            return True
        else
            return (w.mark == v.mark?) False : True
        end if
    end for
end function
color ← 1                           ▷ main function
mark each vertex v ∈ V unvisited
for each vertex v ∈ V marked unvisited do        ▷ start dfs
    if DFS(v)==False then
        return False
    end if
end for
return True
```

Problem 10.5

Require: a given undirected graph $G = (V, E)$ and the color inf. $C = \{c_1 \ldots c_v, c_w \ldots\}$
Ensure: whether this can be a solution of coloring problem

```
for each e(v, w) ∈ G.E do
    if c_v == c_w then
        return False
    end if
end for
return True
```

Problem 10.9

Let I_1 be an instance of Pi_1 and I_2 be an instance of Pi_2
$I_1 \rightarrow_{poly} I_2$ needs $O(n^j)$ time and I_2 can be solved in $O(n^k)$ time.

$\therefore \Pi_2$ can be solved in $O(n^{jk})$ time

Problem 10.19

No, because the complexity of KNAPSACK is not polynomial but psedu-polynomial.

Problem 10.22

1) $NP = P \Rightarrow \exists \Pi \in NPC, \Pi \in P$
$\because NPC \subseteq NP$ and $NP = P$
$\therefore NPC \subseteq P$
$\therefore \exists \Pi \in NPC, \Pi \in P$

2) $\exists \Pi \in NPC, \Pi \in P \Rightarrow NP = P$
$\because \Pi_1 \in NPC \Rightarrow \Pi_1 \in NP$
$\therefore \forall \Pi_2 \in NP, \Pi_2 \propto_{poly} \Pi_1$
$\because \Pi_1 \in P$
$\therefore \forall \Pi_2 \in NP, \Pi_2 \in P$
$\therefore NP = P$

$\therefore NP = P \Leftrightarrow \exists \Pi \in NPC, \Pi \in P$

Optimazing Problem

search version solution: $S(G)$
optimization version solution: $S(G, L)$
The $S(G, L)$ just need to add an operation to find the edge with smallest weight, which can be found in polynomial time.

Hitting Set Problem

Given a graph G, for each edge $e = (u, v) \in G.E$, create a set $S = \{u, v\}$ and add s to H.
Then just prove G has a vertex cover of size b
$\therefore$ VERTEX COVER $\propto_{poly}$ HITTING SET
$\because$ VERTEX COVER $\in NPC$
$\therefore$ HITTING SET $\in NPC$