Übungsblatt 6

Ali Bektas 588063 Julian Kremer 562717
Ruben Dorfner 550204

December 9, 2019

Aufgabe 31

a)

Wir reduzieren VC auf DOMSET.

1. Sei $G = (V, E)$ein **ungerichteter** Graph und sei $G' = (V, E')$ ein **gerichteter** Graph.
2. Für jedes $a, b \in V$ mit $(a, b) \in E$ enthalte $E' \cap = \{(a, b), (b, a)\}$
3. Für jeden Knoten a für den es gilt : $\neg\exists(a, b) \in E$ enthalte $E' \cap = \{(b, a)\}$ für alle $b \in V/\{a\}$

b)

In einem Turniergraph mit n Knoten gibt es $\frac{n \cdot (n-1)}{2}$ Kanten. Im Durchschnitt gibt es also $\frac{(n-1)}{2}$ ausgehende Kanten für jede Knoten. Wir sammeln die DOMSET von einem Graphen wie folgt.

1. Wir nehmen den Knoten , der mindestens $\frac{n-1}{2}$ Kanten hat.
2. Wir nehmen ihn und alle von ihm dominierten Knoten heraus.
3. Wir wiederholen Schritt 1 für den übriggebliebenen Graphen.

Wir nehmen jedes Mal $\underbrace{1}_{\text{gewählter Knoten}} + \left\lceil \frac{(n-1)}{2} \right\rceil$ viele Knoten heraus. Somit ... .Wir sind also in $log(n)$ Schritten fertig.

c)

Da wir wissen , dass wir maximal log(n) Elemente brauchen , um ein Dominating Set für den Graphen zu erstellen ,sei M eine DTM die folgende Arbeitsweise hat:

1. M wählt in aufsteigender Reihenfolge maximal $log(n)$ viele Elemente aus n vielen Elementen
2. Dann versucht sie nachzuweisen dass die gewählte Menge eine Dominating Set für den Grpahen darstellt. Wenn das ihr gelingt hält sie , wenn nicht geht sie weiter.

Somit ist die Anzahl an Rechenschritten :

$$\sum_{i=1}^{\lceil log(n) \rceil} \binom{n}{i} = \frac{n!}{(n-1)!} + \frac{n!}{(n-2)!2!} + \cdots + \frac{n!}{(n - \lceil log(n) \rceil)! \cdot \lceil log(n) \rceil}$$
$$= n + \frac{n * (n-1)}{2} + \cdots + \frac{n \cdot (n-1) \cdot \cdots \cdot (n - \lceil logn \rceil + 1)}{1 \cdot 2 \ldots \lceil logn \rceil}$$
$$\leq n + n \cdot (n-1) + \cdots + n \cdot (n-1) \ldots (n - \lceil logn \rceil + 1) = n^{O(logn)}$$

Die letzte Schranke setzt hingegen $n \geq 2$ voraus. Es ist aber sinnvoll das anzunehmen. Für $n = 1$ sind wir sowieso in O(1) fertig.