$\Rightarrow$ G enthält mind. $\frac{n}{2}$ Knoten vom Grad ≤ 11.

Denn: Ann: nein.

$\Rightarrow$ mind. $\frac{n}{2}$ Knoten sind vom Grad > 11 bzw. dann ≥ 12.

$\Rightarrow \sum_{v \in V} \text{degree}(v) \geq \frac{n}{2} \cdot 12 = 6n$

$\Rightarrow$ ↯ zu (*)

// (*) sagt $\sum_{v \in V} \text{degree}(v) < 6n$!

Betrachte Algorithmus:

Alg. startet mit Markierung der drei begrenzten Knoten.

In diesem Moment gibt es mind. $\frac{n}{2} - 3$ unmarkierte Knoten mit Grad ≤ 11.

Dann wählt der Alg. einen dieser Knoten und markiert ihn zusammen mit seinen höchstens 11 Nachbarn.

Das wird solange wiederholt bis es keine unmarkierten Knoten mit Grad ≤ 11 mehr gibt.

Deshalb werden während jeder Iteration höchstens 12 Knoten markiert.

Es gibt also mindestens $\left\lceil \frac{1}{12} \left(\frac{n}{2} - 3 \right) \right\rceil$

Während jeder Iteration wird ein Knoten zu I hinzugefügt.

Ergebnis: $\forall$ Triang. G gilt: G enthält unabh. Knotenmenge I, so dass

1) $|I| \geq \left\lceil \frac{1}{12} \left(\frac{n}{2} - 3 \right) \right\rceil$
2) $\forall v \in I$: degree(v) ≤ 11
3) I enthält keinen der drei Randknoten.
4) I kann in Zeit $O(n)$ berechnet werden.

⟶ 3.4.5.9 Lemma: Sei I eine unabh. Knotenmenge von G gemäß 3.4.5.8.

Sei G' der Graph, der durch Entfernen von I aus G entsteht

$\Rightarrow$ 1). Jede Fläche von G' ist ein 1-faches Polygon mit maximal 11 Knoten

2). Die äußeren Flächen (Dreiecke) von G und G' sind gleich.

⟶ 3.4.5.10 Alg. zu Konstruktion der Datenstruktur D zur Darstellung der Folge $S_1 .. S_k$:

Datenstruktur D: Knoten + Pointer.

Knoten v speichert ein Dreieck t $v = n(t)$.

Triangulierung S: Pkte, Kanten, Flächen

$n_i = |S_i|$ = # der Pkte.

1. $i \leftarrow 1$, $S_1 \leftarrow G$.
2. $\forall$ Dreiecke t in G do
3. erzeuge einen Knoten n(t)
4. od.
5. while $|S_i| > 3$ do
6. berechne unabh. Knotenmenge I in S_i mit mind. $\left\lceil \frac{1}{12} \left(\frac{n}{2} - 3 \right) \right\rceil$ Knoten, die nicht am Rand liegen und maximal Grad 11 haben.
7. Entferne die Knoten von I und ihre angegrenzten Kanten aus S_i
8. Trianguliere den entstandenen Graphen und nenne das Resultat S_{i+1}.
9. $\forall$ Dreiecke t von S_{i+1}, die nicht in S_i enthalten sind (d.h. neue Dreiecke) do
10. erzeuge Knoten n(t)
11. $\forall$ Knoten n(t') mit $t' \in S_i$ und t' schneidet t do
12. erzeuge einen Pointer $n(t) \longrightarrow n(t')$
13. od
14. od
15. $i \leftarrow i+1$
16. od