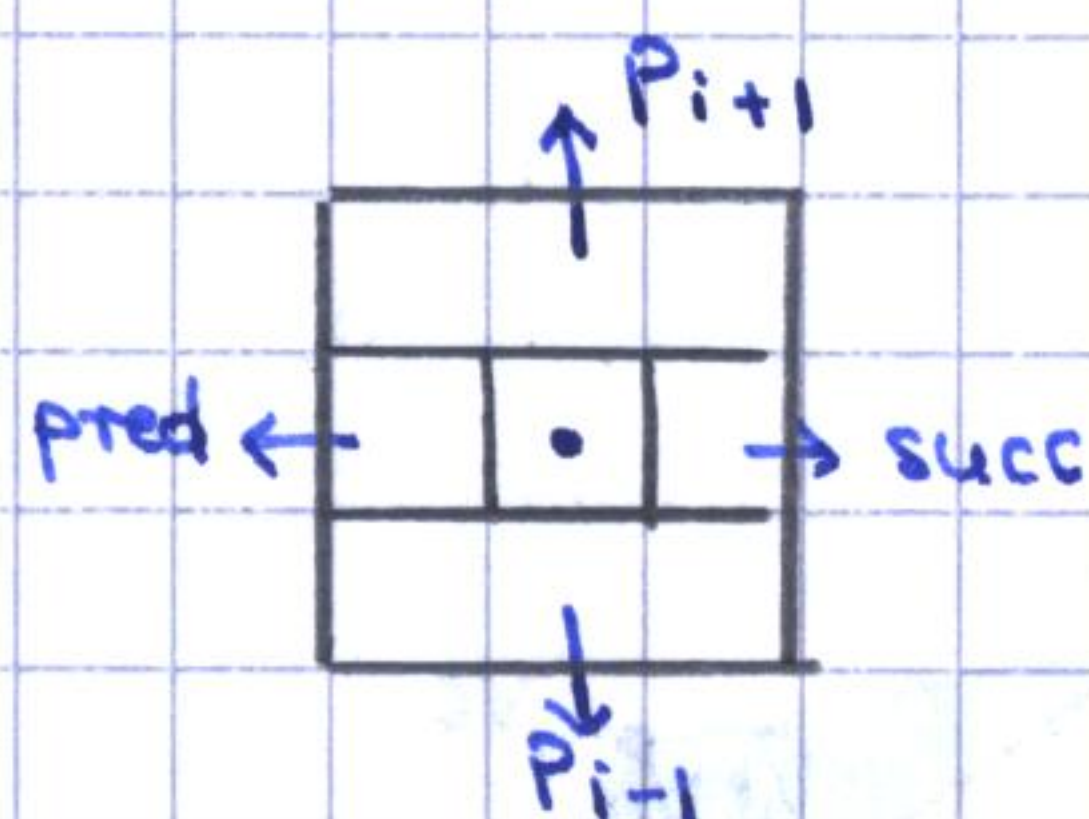


### 2.2.2. Darstellung im Rechner:

- $P_i, 0 \leq i \leq k$  als doppelt verkettete Liste der Ecken + jede Ecke von  $P_i$  zeigt zusätzlich auf ihre Kopie in  $P_{i+1}$  und umgekehrt

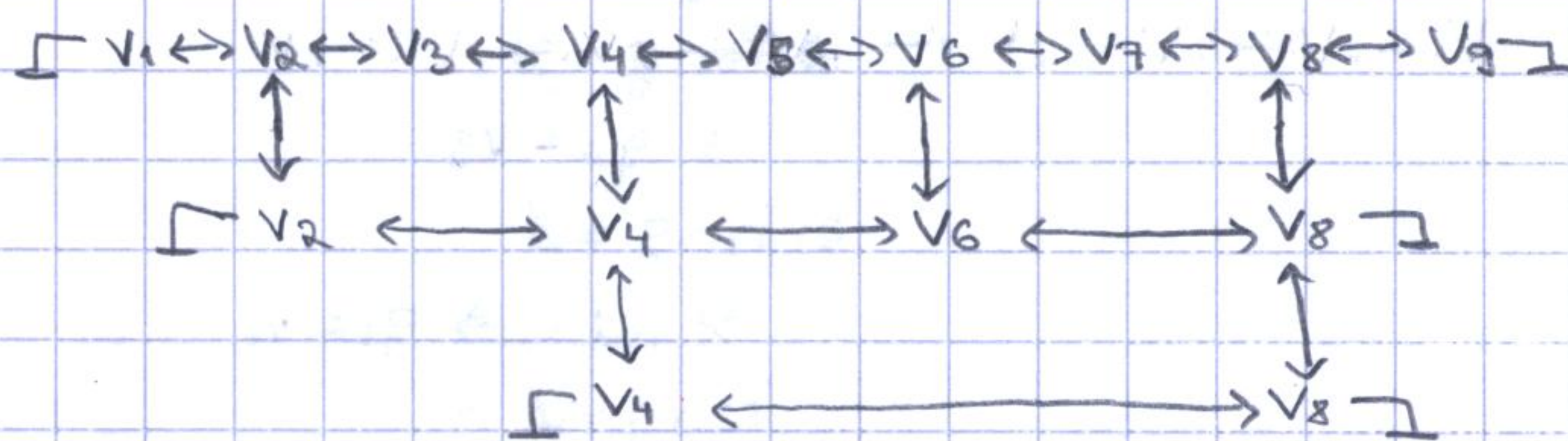
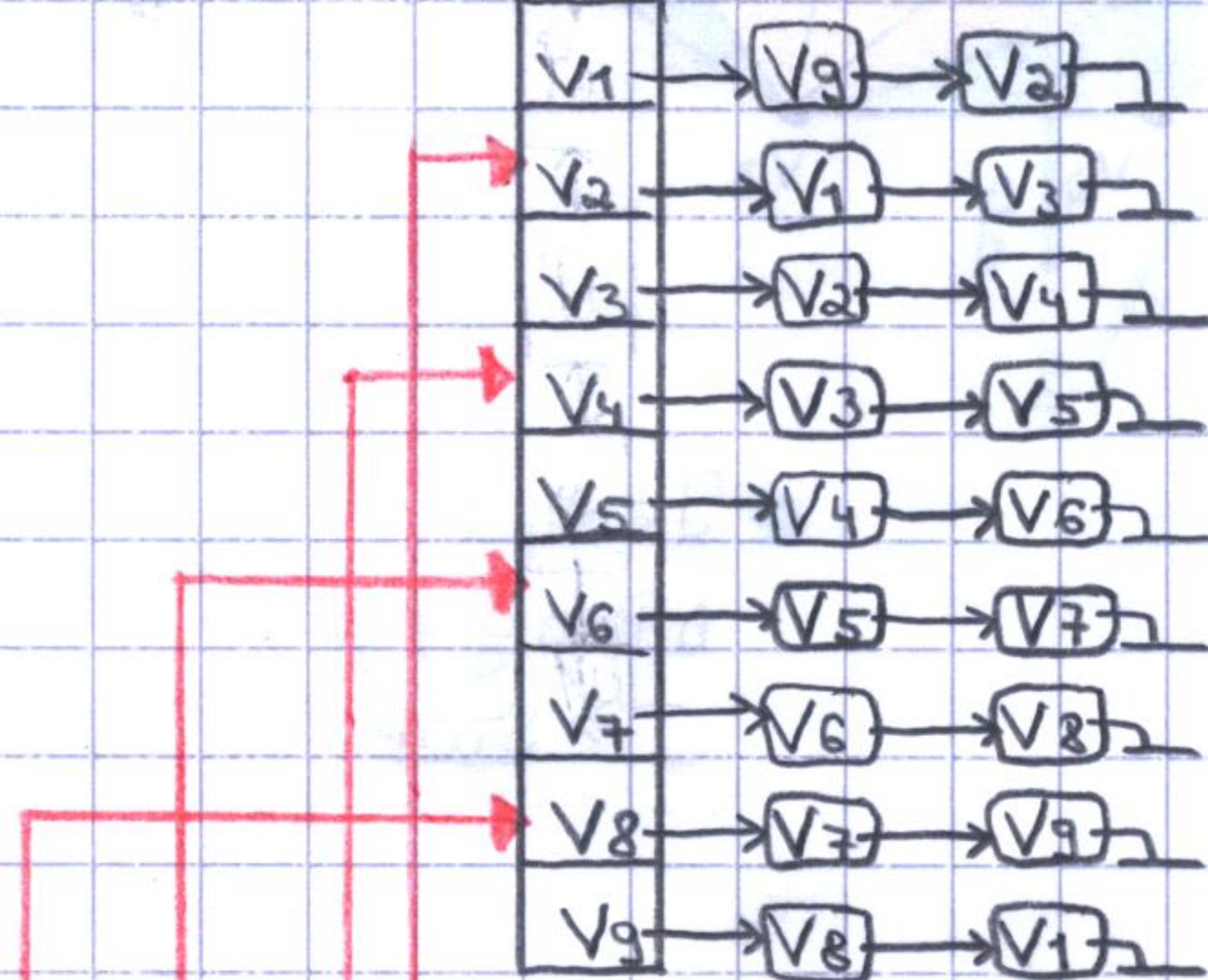


- explizite Speicherung des Kantenbaumes  
→ Dynamisierung (Ecken einfügen / löschen)  
(wie bei Suchbäumen)

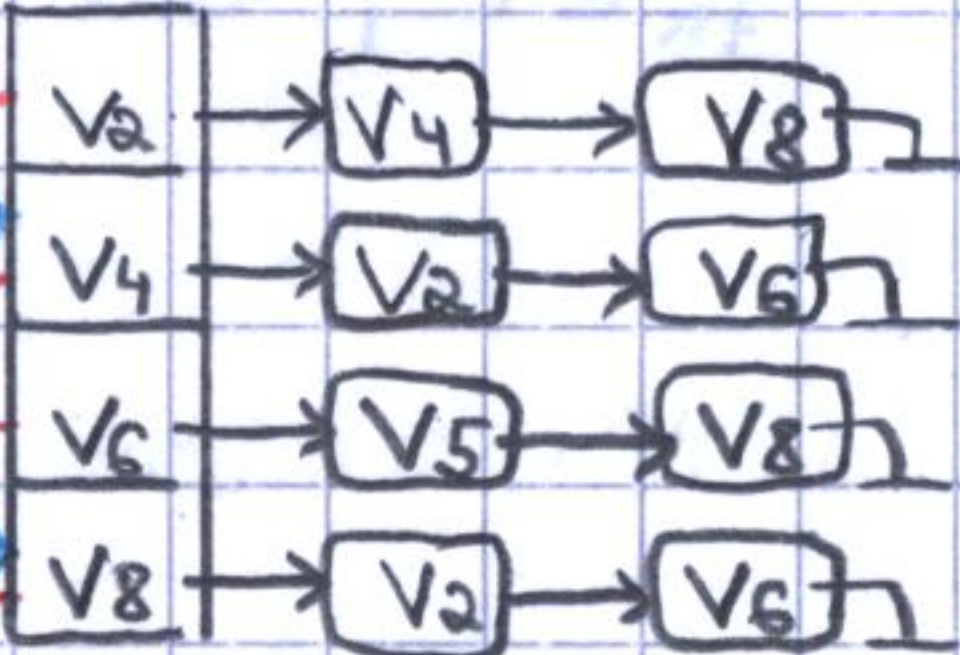
### 2.2.3 Beispiel:

Sei  $P$  gegeben wie in 2.1.9.

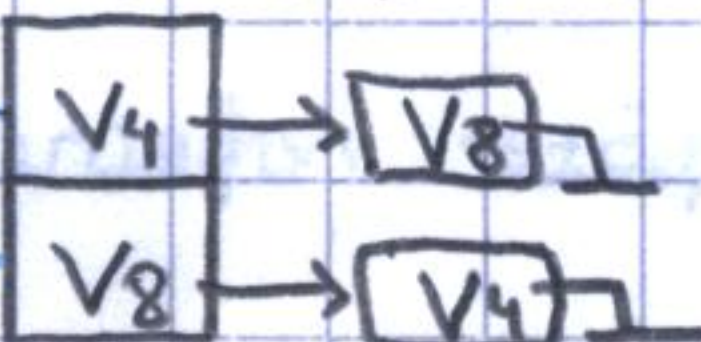
$P_2$  als verkettete Liste:



$P_1$  als verkettete Liste:



$P_0$  als verkettete Liste:



### 2.2.4. Anwendung: Schnitt mit einer Geraden:

#### 2.2.4.1. Ziel:

Geg: Hierarchische Darstellung  $P_0 \dots P_k$  eines konvexen Polygons  $P$  und eine Gerade  $g$ .

Ausgabe:  $P \cap g$

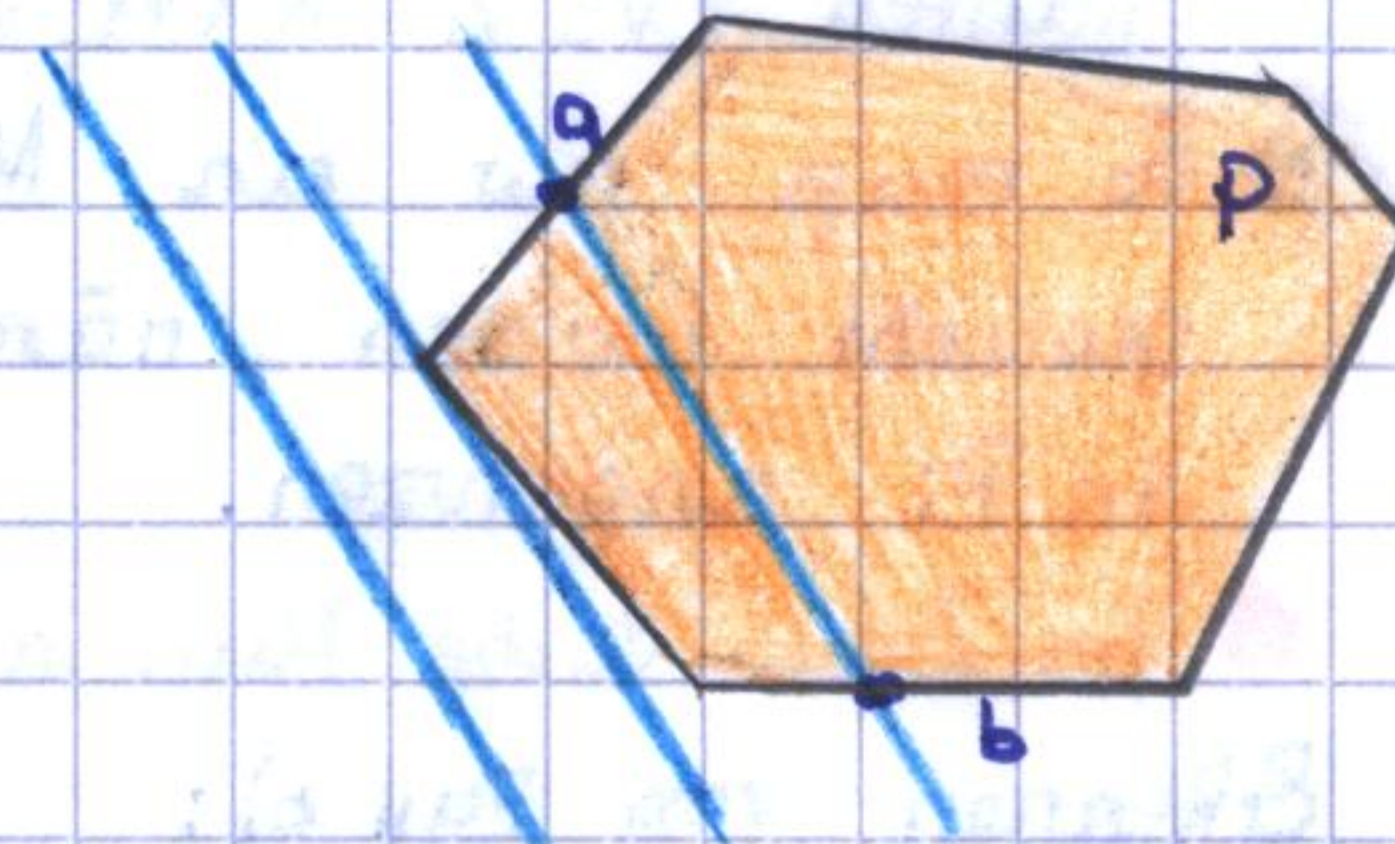
Konvexität  $\Rightarrow P \cap g =$  Strecke

$\Rightarrow$  Es genügt Schnittpunkte  $a, b$  mit den Randsegmenten zu berechnen.

Fälle: 1)  $a \neq b$  (allg. Fall, Ecken möglich)

2)  $a = b$  (eine Ecke)

3) ex. nicht!, Schnitt =  $\emptyset$



#### 2.2.4.2. Idee für Algorithmus:

1) Schnitt mit  $P_0$ :

2 Fälle: a)  $P_0 \cap g = \emptyset$

b)  $P_0 \cap g \neq \emptyset$

konst. Operationen, weil  $|P_0| \leq 4$

Ann: wir haben Fall a)

$\Rightarrow$  sei  $g_0 \in P_0$  mit minimalen Abstand zu  $g$

Im Fall b  $\rightarrow$  STOP