Aufgabe 2 (Quick-Select)

Beschreibung des Selection-Problems

Gegeben ist eine natürlich Zahl **k** und eine Menge S mit **n** Elementen x_1, x_2, ..., x_n aus einer total geordneten Menge. **Ziel**: Finde das k-kleinste Element dieser Menge.

Langsame Lösung: Man könnte die Menge sortieren und dann das k-te Element indizieren.

k = 3 7 4 9 6 2 → 2 4 6 7 9

Schnellere Lösung: Verwende den zufallsgesteuerten Algorithmus **Quick-Select**, der nach dem Prune-and-Search-Paradigma arbeitet. Idee:

- **Prune**: wähle zufällig ein Element **x** (genannt pivot) aus der Menge S und teile S in folgende drei Mengen L, E und G auf:
  - L – Elemente kleiner als x
  - E – Elemente gleich x
  - G – Elemente größer als x
- **Search**: abhängig von k kann man jetzt entscheiden, in welcher dieser drei Mengen x liegt

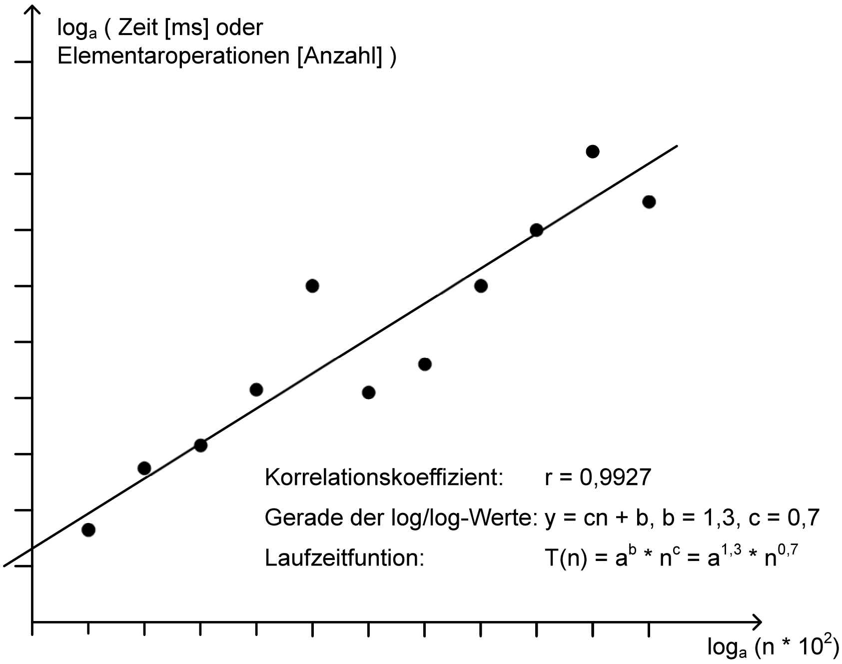
a) Erklären Sie die **rekursive Idee** von **Quick-Select** und formulieren Sie den Algorithmen rekursiv als **Pseudocode**.

b) Programmieren Sie den Algorithmus rekursiv. Die Größe der Menge S und die Zahl k sollen interaktiv eingegeben werden. Sellen Sie den Ablauf des Algorithmus dar und geben Sie die Mengen L, E und G nach jedem Prune & Search-Schritt aus.

c) Analysieren Sie Quick-Select

1. **Theoretische Analyse** der asymptotischen Laufzeit
2. **Experimentelle Analyse** mit dem Powertests. Geben Sie den Korrelationskoeffizienten r, die log/log-Gerade und die Laufzeitfunktion t(n) an.

Stellen Sie die **Ergebnisse** der experiementellen Analyse wie in der Vorlesung besprochen **grafisch** dar. Ein Darstellungsbeispiel steht auf der nächsten Seite.



● Logarithmierte (Basis a) Mess- bzw. Zählwerte

—— Durch lineare Regression ermittelte Gerade $y = cn + b$

a: Basis des Logarithmus
b: Schnittpunkt der Geraden mit der y-Achse
c: Steigung der Geraden