

Handout: Gamble Search Algorithmus

Einleitung

Der **Gamble Search Algorithmus** ist eine Variation des klassischen **Binary Search**. Im Gegensatz zum normalen Binary Search, bei dem immer der mittlere Punkt gewählt wird, wählt der Gamble Search einen **zufälligen Punkt** innerhalb des Suchbereichs, um diesen zu durchsuchen. Dieser zufällige Pivot macht den Algorithmus dynamischer und bietet unter bestimmten Bedingungen Vorteile.

Merkmale von Gamble Search

Der Algorithmus zeichnet sich durch eine zufällige Pivot-Auswahl aus, wobei der Pivot zufällig innerhalb des Suchbereichs bestimmt wird. Die Suche erfolgt in sortierten Daten, ähnlich wie beim Binary Search. Die durchschnittliche Zeitkomplexität liegt bei $O(\log n)$, jedoch mit höherer Varianz durch die zufällige Pivotwahl.

Vor- und Nachteile

Die Vorteile liegen in der Unvorhersehbarkeit durch zufällige Pivotwahl und möglichen Optimierungen bei ungleichmässiger Datenverteilung. Nachteile sind die höhere Varianz in der Laufzeit und ein leicht erhöhter Overhead im Vergleich zum klassischen Binary Search.

Anwendungen & Technische Umsetzung

Der Algorithmus eignet sich besonders für Anwendungen, die von Unvorhersehbarkeit profitieren, wie Spiele oder Glücksspiele. Die Implementierung erfolgt in Python unter Nutzung von Standardfunktionen und der random-Bibliothek. Die Funktionalität wird durch Unit-Tests mit pytest sichergestellt.

Fazit

Gamble Search bietet als Alternative zum Binary Search durch seinen Zufallsfaktor eine flexible, wenn auch weniger vorhersehbare Performanz, die in speziellen Anwendungsfällen vorteilhaft sein kann.