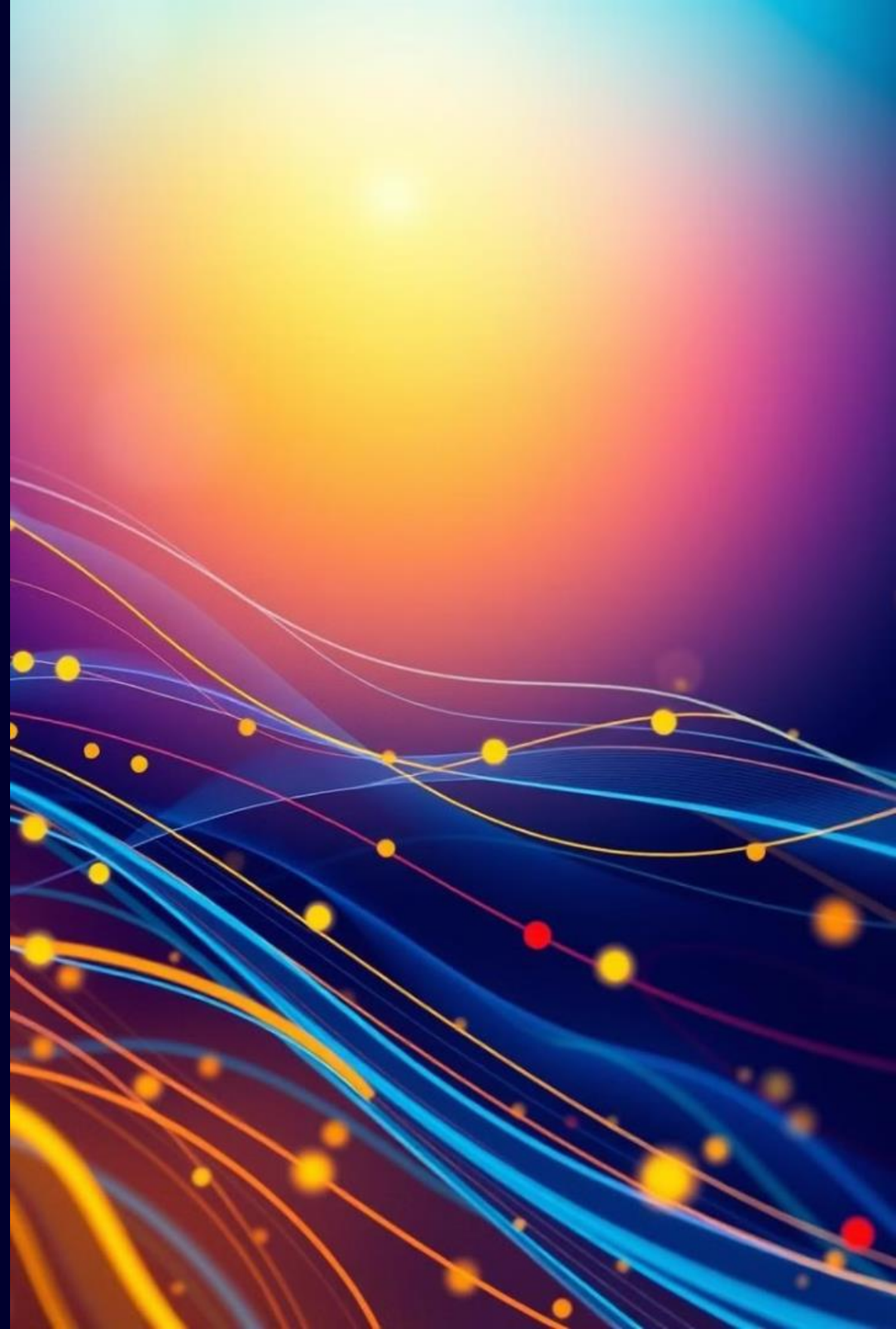
Algorytmy wyszukiwania i porządkowania

Zapraszamy do eksploracji fascynującego świata algorytmów wyszukiwania i porządkowania. Przyjrzymy się różnym technikom i ich zastosowaniom w informatyce.

JK **by Jakub Kam**



Algorytmy wyszukiwania

1 **Cel**

Znalezienie konkretnego elementu w zbiorze danych, np. liczby w liście.

2 **Efektywność**

Czas potrzebny do znalezienia elementu zależy od algorytmu i wielkości zbioru.

3 **Rodzaje**

Istnieje wiele algorytmów, np. liniowy, binarny, hashowanie.

4 **Zastosowania**

Wyszukiwanie informacji w bazach danych, wyszukiwarki internetowe.

Algorytm wyszukiwania liniowego

1. Krok 1

   Rozpocznij od początku zbioru danych.

2. Krok 2

   Porównaj aktualny element ze szukanym.

3. Krok 3

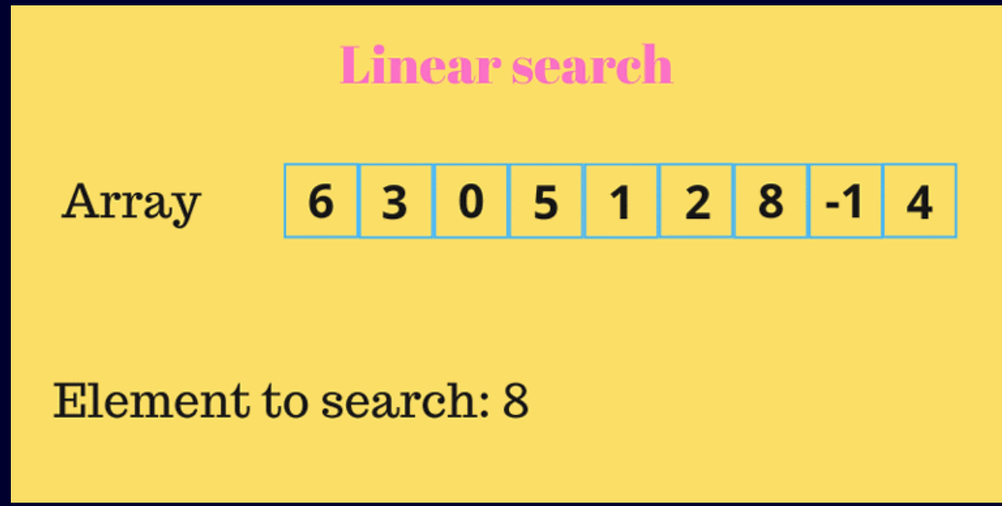
   Jeśli elementy są identyczne, zakończ wyszukiwanie.

4. Krok 4

   Przesuń się do następnego elementu i powtórz kroki 2-3.

5. Krok 5

   Jeśli przejrzysz cały zbiór bez znalezienia elementu, zakończ wyszukiwanie.



Algorytm wyszukiwania binarnego

Zasada

Działanie na posortowanym zbiorze danych. W każdym kroku algorytm dzieli zbiór na pół i porównuje element środkowy ze szukanym.

Złożoność

Efektywny, logarytmiczny czas wyszukiwania, znacznie szybszy niż wyszukiwanie liniowe.

Przykład

Wyszukiwanie słowa w słowniku, gdzie słowa są posortowane alfabetycznie.

Algorytmy porządkowania

Cel

Uporządkowanie zbioru danych w określonym porządku, np. od najmniejszego do największego.

Metody

Istnieje wiele metod sortowania, każda z różną złożonością i wydajnością.

Zastosowania

Sortowanie danych w bazach danych, wyszukiwanie informacji, algorytmy uczenia maszynowego.

Rodzaje

Sortowanie bąbelkowe, sortowanie przez wstawianie, sortowanie przez scalanie, sortowanie szybkie.

Algorytm sortowania bąbelkowego

1

Krok 1

Porównaj dwa sąsiednie elementy.

2

Krok 2

Zamień elementy, jeśli są w złej kolejności.

3

Krok 3

Powtórz kroki 1-2 dla wszystkich par sąsiednich elementów.

4

Krok 4

Powtórz kroki 1-3 aż do momentu, gdy zbiór będzie posortowany.

Algorytm sortowania przez wstawianie

Krok	Opis
1	Utwórz dwie części: posortowaną (z jednym elementem) i nieposortowaną.
2	Wybierz pierwszy element z nieposortowanej części.
3	Porównaj go z elementami w posortowanej części.
4	Wstaw element w odpowiednie miejsce w posortowanej części.
5	Powtórz kroki 2-4 dla pozostałych elementów z nieposortowanej części.

Algorytm sortowania przez scalanie

Dzielenie

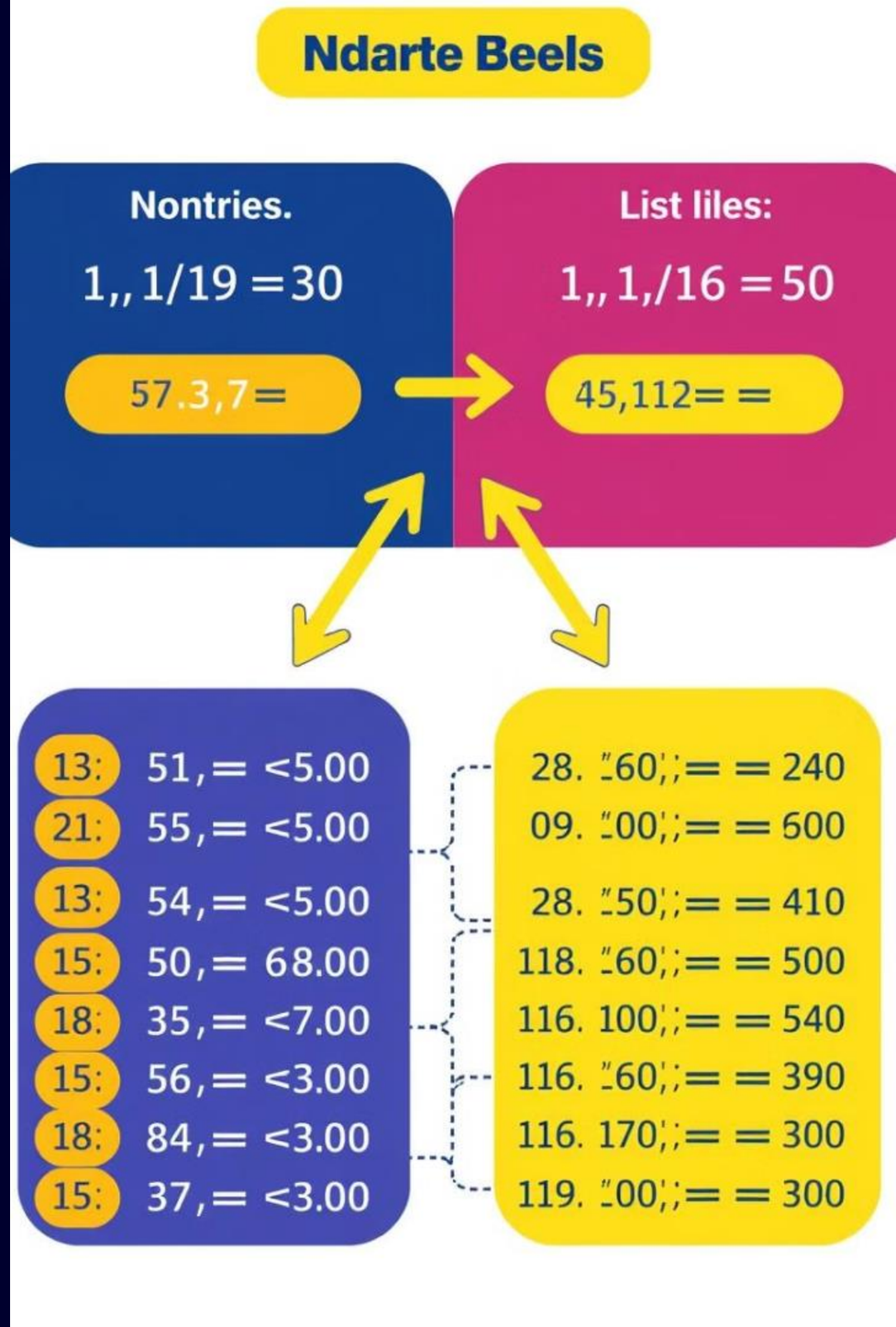
Zbiór danych jest rekurencyjnie dzielony na mniejsze podzbiory.

Sortowanie

Podzbiory są sortowane za pomocą dowolnej metody sortowania.

Scalanie

Posortowane podzbiory są scalane w jeden posortowany zbiór.



Algorytm szybkiego sortowania

1

Krok 1

Wybierz element jako pivot.

2

Krok 2

Podziel zbiór na dwie części: mniejsze i większe od pivot.

3

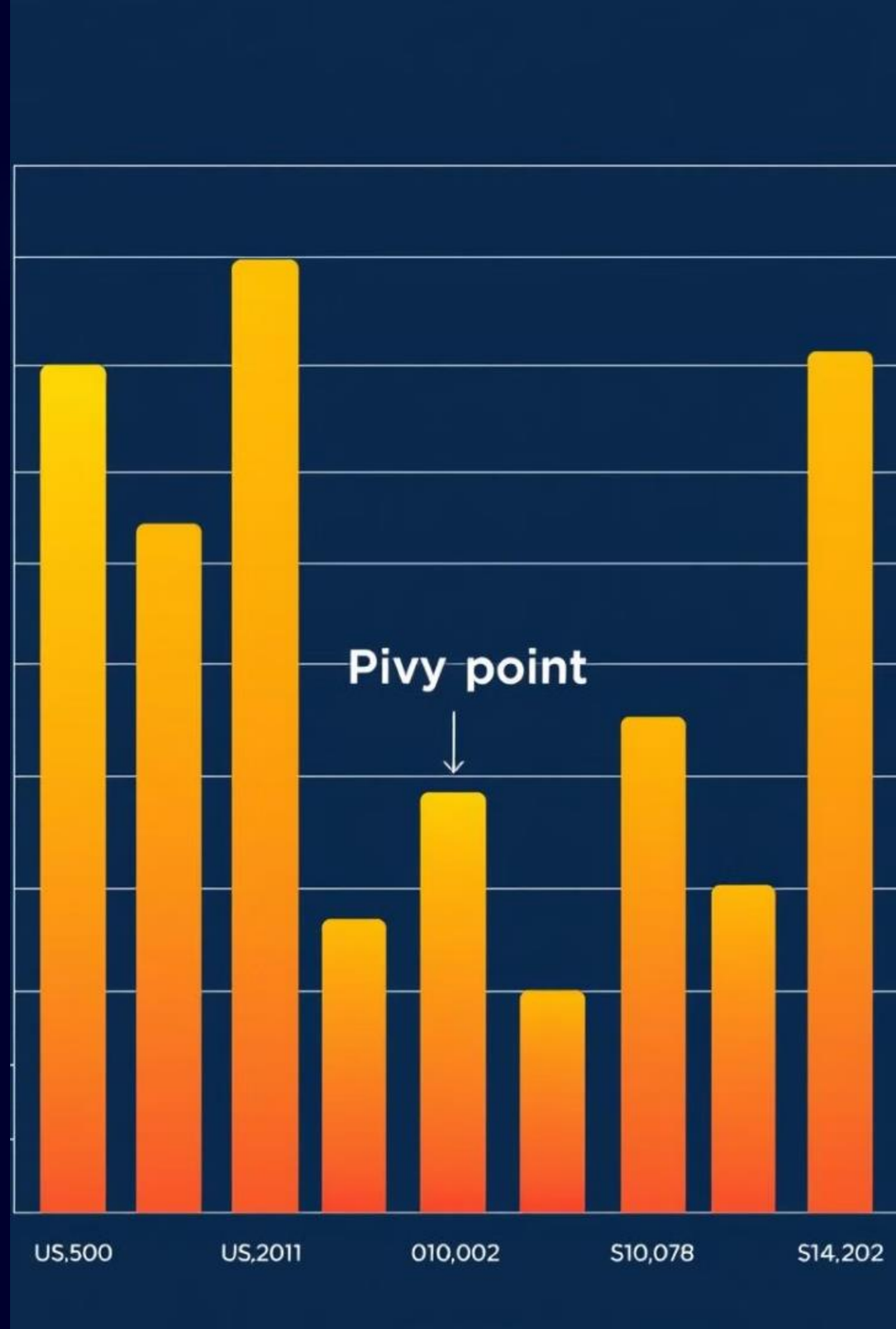
Krok 3

Rekurencyjnie sortuj obie części.

4

Krok 4

Połącz posortowane części.



Podsumowanie i wnioski

Algorytmy wyszukiwania i porządkowania są podstawowymi narzędziami w informatyce. Zrozumienie ich działania pozwala na tworzenie bardziej wydajnych i efektywnych programów. Wybór odpowiedniego algorytmu zależy od specyfiki problemu i dostępnych zasobów.