Efektywność algorytmów

Algorytmika

- **Algorytmika** to dział informatyki zajmujący się poszukiwaniem, konstruowaniem i badaniem własności algorytmów, w kontekście ich przydatności do rozwiązywania problemów obliczeniowych za pomocą komputerów.
- Nazwa ta została wprowadzona przez izraelskiego informatyka Davida Harela w tytule jego książki o algorytmach: *Algorithmics. The Spirit of Computing.*
- Zadania algorytmiki:
  - projektowanie algorytmów i szacowanie ich złożoności obliczeniowej;
  - tworzenie ogólnych metod rozwiązujących pewne grupy zadań obliczeniowych;
  - badanie właściwości pewnych klas problemów obliczeniowych;
  - projektowanie struktur danych i operacji na tych strukturach;
  - analizowanie stanu po modyfikacjach struktur danych;
  - badanie złożoności obliczeniowej operacji na strukturach danych.

Od zadania do rozwiązania

- Analiza zadania
- Zaprojektowanie algorytmu
- Szacowanie złożoności
- Optymalizacja
- Implementacja i kompilacja
- Przygotowanie danych
- Uruchomienie i odczytanie wyników

Zadanie obliczeniowe

- **Problem obliczeniowy** to zadanie, które może być rozwiązane za pomocą komputera lub innej maszyny liczącej.
- **Specyfikacja** zadania:
  - zbiór **danych** wyjściowych
  - warunki jakie ma spełniać **wynik**
  - **zasoby** komputerowe do dyspozycji obliczeń (czas, pamięć, komunikacja, itp)
- Problem obliczeniowy to funkcja, która przekształca zbiór danych wejściowych na zbiór danych wyjściowych.

Algorytm

- **Algorytm** to sposób na rozwiązanie zadania obliczeniowego (precyzyjnie zdefiniowany ciąg czynności, koniecznych do przeprowadzenia systemu z określonego stanu początkowego do zadanego stanu końcowego).
- Słowo "algorytm" pochodzi od nazwy "Algoritmi" – zlatynizowanej wersji nazwiska Abu Abdullaha Muhammada ibn Musy al-Chuwarizmiego, matematyka perskiego z IX wieku.
- Może istnieć wiele algorytmów rozwiązujących jeden problem obliczeniowy.
- Zapis algorytmu:
  - wskazówka
  - lista kroków
  - schemat blokowy
  - pesudokod

Złożoność obliczeniowa algorytmu

- **Złożoność obliczeniowa** algorytmu to jego zapotrzebowanie na pamięć w zapotrzebowanie na zasoby komputerowe (czas, pamięć, komunikacja, itp) potrzebne realizacji obliczeń.
- Złożoność obliczeniową algorytmu szacujemy od góry: ile zasobów z najgorszym przypadku wystarczy do przeprowadzenia obliczeń?
- Złożoność obliczeniową problemu szacujemy z dołu: ile zasobów zużyje dowolny algorytm rozwiązujący określony problem w najgorszym przypadku?
- Złożoność obliczeniowa jest funkcją rozmiaru danych wejściowych.

Złożoność obliczeniowa algorytmu

- **Złożoność pamięciowa** to zapotrzebowanie algorytmu na pamięć, którą liczymy w **komórkach**:
  - **dane wejściowe** dostarczone do algorytmu **nie** wchodzą do kosztów pamięciowych;
  - wywołania funkcji generują koszty pamięciowe – w szczególności **wywołania rekurencyjne**.
- **Złożoność czasowa** to zapotrzebowanie algorytmu na czas, którą liczymy za pomocą **operacji dominujących**:
  - operacja dominująca to proste obliczenie, działające w czasie stałym, które ma najistotniejszy wpływ na liczbę kroków algorytmu.

Złożoność obliczeniowa algorytmu

- Złożoność obliczeniowa zwykle nie zależy wyłącznie od rozmiaru danych, ale może się znacznie różnić dla danych wejściowych o identycznym rozmiarze. Często stosowanymi kryteriami są:
  - **złożoność pesymistyczna** – rozpatrywanie przypadków najgorszych;
  - **złożoność oczekiwana** – zastosowanie określonego sposobu uśrednienia wszystkich możliwych przypadków;
  - złożoność optymistyczna – rozpatrywanie przypadków najprostszych (rzadko stosowane kryterium).

Asymptotyka

- Złożoność obliczeniowa jest zwykle bardzo skomplikowaną funkcją, dlatego szacuje się ją z dokładnością do pewnego stałego czynnika (czasem z dokładnością do nieistotnego składnika) za pomocą operatorów asymptotycznych.
- Założenia:
  - rozpatrujemy funkcje $\mathbb{N} \rightarrow \mathbb{R}_+$
  - funkcja graniczna to g(n)
- Notacja dużego **O**, **Ω** i **Θ** została zaproponowana po raz pierwszy w roku 1894 przez niemieckiego matematyka Paula Bachmanna. W późniejszych latach spopularyzował ją w swoich pracach Edmund Landau, niemiecki matematyk, stąd czasem nazywana jest notacją Landaua.

Asymptotyka

- Ograniczenie górne wyznaczone przez funkcję g(n):
  O(g(n)) $= \{ f(n) : \exists\, c>0\ \exists\, n_0\ \ n \geq n_0 : f(n) \leq c \cdot g(n) \}$
- Ograniczenie dolne wyznaczone przez funkcję g(n):
  Ω (g(n)) $= \{ f(n) : \exists\, c>0\ \exists\, n_0\ \ n \geq n_0 : c \cdot g(n) \leq f(n) \}$
- Ograniczenie dokładne wyznaczone przez funkcję g(n):
  Θ (g(n)) $= O(g(n)) \cap \Omega\,(g(n))$
- Asymptotyczna równość funkcji $f(n) \sim g(n)$: