Metaheurystyki

Tadeusz Puźniakowski

PJATK

16 marca 2024

Spis treści

Plan na dziś

- O przedmiocie
- Zarys tematów omawianych
- Przypomnienie o złożoności obliczeniowej (wersja dla praktyków)

O zajęciach

Wykład

Wykład jest przygotowaniem do ćwiczeń. Jest też wstępem teoretycznym do tematu. Jest egzamin.

Laboratoria

Ćwiczenia są z bieżącego tematu omawianego na wykładzie. Zaliczenie jest na podstawie punktów z zajęć.

Spis treści

Algorytm

Sekwencja instrukcji służąca do osiągnięcia określonego celu.

Funkcja heurystyczna - heuristic function

Funkcja służąca do szacowania/wskazywania korzystnego kierunku prowadzenia obliczeń.

Heurystyka

Konkretna metoda prowadzenia obliczeń która nie daje gwarancji znalezienia najlepszego rozwiązania.

Metaheurystyka

Ogólna metoda prowadzenia obliczeń która pozwala na znajdywanie rozwiązań bez gwarancji znalezienia najlepszego rozwiązania.

Metaheurystyka a heurystyka

Metaheurystyka z ustalonymi parametrami staje się heurystyką.

Złożoność problemu

Ilość zasobów (pamięć/procesor) wymagana do wykonania zadania.

Asymptotyczne tempo wzrostu

Miara złożoności która określa jak zachowuje się algorytm dla danego problemu.
Określa ona funkcję według której rośnie złożoność w zależności od rozmiaru danych.

Notacja asymptotyczna

Zapis określający asymptotyczne tempo wzrostu. Najczęściej stosuje się O, o, Ω, oraz Θ

Notacja asymptotyczna

f jest co najwyżej rzędu g : $f(n) \in \mathrm{O}(g(n))$

Gdy istnieją takie $n_0 > 0$, $c > 0$, że $\forall n \geqslant n_0 : f(n) \leqslant c \cdot g(n)$

f jest niższego rzędu niż g : $f(n) \in \mathrm{o}(g(n))$

$\forall c > 0$ istnieje stała $n_0 > 0$, taka że $\forall n \geqslant n_0 : f(n) < c \cdot g(n)$

f jest co najmniej rzędu g : $f(n) \in \Omega(g(n))$ (omega)

istnieją takie stałe $n_0 > 0$, oraz $c > 0$ że: $\forall n \geqslant n_0 : f(n) \geqslant c \cdot g(n)$

f jest dokładnie rzędu g : $f(n) \in \Theta(g(n)$ (theta)

istnieją takie stałe $n_0 > 0$, oraz $c_1 > 0$, $c_2 > 0$,że:
$\forall n \geqslant n_0 : c_1 \cdot g(n) \leqslant f(n) \leqslant c_2 \cdot g(n)$

Klasy złożoności

Klasa złożoności

Zbiór problemów o tej samej złożoności asymptotycznej

Klasa P: deterministic polynomial – deterministycznie wielomianowy

Problem dla którego można znaleźć rozwiązanie w czasie wielomianowym.

Problem NP: nondeterministic polynomial – niedeterministycznie wielomianowy

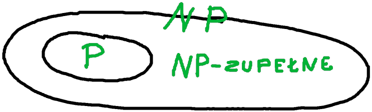
Problem którego rozwiązanie można zweryfikować w czasie wielomianowym.

Mamy też problemy NP zupełne i NP trudne - wyjaśnienie wykracza poza zakres dzisiejszego wykładu.

Klasy złożoności

Trudność problemu

Problem jest łatwy, jeśli każda ścieżka przeszukiwania dziedziny problemu wzdłuż nierosnącej (niemalejącej) funkcji celu prowadzi do rozwiązania. Długość zależy wielomianowo od ilości danych.



Źródła

- „Algorytmy genetyczne + struktury danych = programy ewolucyjne”, Z. Michalewicz
- „Algorytmy genetyczne i ich zastosowania”, D. E. Goldberg
- ”Wykłady z algorytmów ewolucyjnych”, J. Arabas