Olaf Krawczyk 21816

Pn TP 9:15

09.01.2017

Projektowanie Efektywnych Algorytmów

# Programowanie dynamiczne dla problemu plecakowego

Prowadzący: dr inż. Jarosław Mierzwa

**Cel ćwiczenia:**

Celem ćwiczenia było stworzenie programu rozwiązującego problem plecakowy za pomocą programowania dynamicznego. Problem plecakowy jest jednym z najbardziej rozpowszechnionych problemów optymalizacyjnych. W problemie tym mamy daną listę dostępnych przedmiotów, z której każdy przedmiot ma określoną wagę i wartość oraz plecak ustalonej pojemności. Naszym celem wybranie takiego podzbioru przedmiotów w którym sumaryczna waga nie przekracza pojemności plecaka oraz łączna wartość przedmiotów jest największa.

**Opis programu:**

Lista przedmiotów wczytywana jest do programu z pliku podanego przez użytkownika. Wczytane przedmioty zostają zapisane w wektorze zawierającym obiekty klasy Item. Każdy obiekt klasy Item posiada atrybuty przechowujące informacje o wadze oraz wartości danego przedmiotu. Dodatkowo klasa Item udostępnia szereg metod pozwalających np. na dostęp do atrybutów, co umożliwia ich porównywanie i tym samym sortowanie na podstawie np. współczynnika wartość/waga.

Testy algorytmu zostały przeprowadzone dla problemów o rozmiarach 10, 50, 100, 200, 300, 400, 500 oraz stałego rozmiaru plecaka. Dla każdego rozmiaru wykonanych zostało 100 powtórzeń algorytmu. Dla każdej ze 100 iteracji generowany był losowy problem o zadanym rozmiarze instancji. Do pomiaru czasu wykorzystana została biblioteka windows.h oraz funkcje performanceCountStart i performanceCountEndPoniżej.

**Wyniki:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| N | 10 | 50 | 100 | 200 | 300 | 400 | 500 |
| czas [ms] | 0.0260883 | 0.114656 | 0.229197 | 0.413596 | 0.612374 | 0.774466 | 1.05128 |

Tabela Wyniki pomiarów dla DP

Rysunek Czas wykonania w zależności od rozmiaru problemu DP

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| N | 5 | 10 | 15 | 20 | 25 | 27 |
| czas [ms] | 0.00159475 | 0.018908 | 0.517232 | 15.2405 | 495.426 | 2147.1 |

Tabela Wyniki pomiarów dla Brute Force

Rysunek Czas wykonania w zależności od rozmiaru problemu BF

Powyższy wykres pokazuje, że czas znalezienia korzystnego rozwiązania jest ściśle związany z rozmiarem problemu. Algorytm realizujący programowanie dynamiczne radzi sobie z znacznie większymi instancjami problemu niż Brute Force. Czas wykonania DP również jest zauważalnie krótszy niż Brute Force.

**Bibliografia:**

Z. Michalewicz, D. Fogel „Jak to rozwiązać, czyli nowoczesna heurystyka”