Olaf Krawczyk 21816

Pn TP 9:15

12.12.2016

Projektowanie Efektywnych Algorytmów

# Metoda podziału i ograniczeń dla problemu plecakowego

Prowadzący: dr inż. Jarosław Mierzwa

**Cel ćwiczenia:**

Celem ćwiczenia była implementacja algorytmu podziału i ograniczeń dla problemu plecakowego. 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. Sortowanie tego typu jest kluczowe przy wyznaczaniu granicy kolejnych rozwiązań. Wyznaczanie granicy opera się na zachłannym dobieraniu kolejnych elementów na podstawie wspomnianego wcześniej kryterium. Program dobiera kolejne możliwe przedmioty, a w przypadku, gdy waga aktualnie dobranych przedmiotów przekracza pojemność plecaka granica powiększana jest o ułamek wartości ostatniego badanego elementu. Drzewo rozwiązań przeszukiwane jest za pomocą metody przeszukiwania wszerz. Kolejne rozwiązania reprezentowane są przez struktury Node, które przechowują informacje o rozmiarze, wartości danego rozwiązania oraz historię wybranych elementów.

Testy algorytmu zostały przeprowadzone dla problemów o rozmiarach 10, 50, 80, 150, 200, 230, 250, 260 oraz stałego rozmiaru plecaka. Dla każdego rozmiaru wykonanych zostało 100 powtórzeń algorytmu. Dla każdej ze 100 operacji 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 | 80 | 150 | 200 | 230 | 250 | 260 |
| czas [μs] | 0,027403 | 0,313976 | 1,11195 | 8,42711 | 55,3029 | 78,0997 | 81,596 | 137,577 |

Tabela Wyniki pomiarów

Rysunek Czas wykonania w zelażności od rozmiaru problemu

Powyższy wykres pokazuje, że czas znalezienia korzystnego rozwiązania jest ściśle związany z rozmiarem problemu. Zależność ta jest nieliniowa. Czas wykonania algorytmu zależności od implementacji może ulec zmianie.

**Bibliografia:**

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