

Problem plecakowy

200439

25 maja 2014

Problem plecakowy to jedno z najczęściej poruszanych zagadnień optymalizacji. Nazwa tego problemu wzięła się z analogii do rzeczywistej sytuacji pakowania plecaka. Należy go zapakować w ten sposób, by pomieścić w nim rzeczy o łącznej największej możliwej wartości. Przy tym nie wolno przekroczyć największej dopuszczalnej wagi plecaka, oraz przedmioty nie mogą być rozdzielane na mniejsze części. Istnieje kilka odmian tego problemu, w tym zadaniu został zrealizowany przypadek, gdy do dyspozycji mamy ustaloną ilość różnych rodzajów przedmiotów w nieograniczonej liczbie sztuk. Każdy przedmiot ma swoją nazwę, wagę, oraz wartość (cenę). Jak więc ustalić zestaw przedmiotów do spakowania do plecaka, które jak najdokładniej go wypełnią, jednocześnie dając największą wartość zapakowanych przedmiotów?

Jest to problem, którego rozwiązanie wielomianowe nie jest znane - problem NP trudny. Istnieje kilka możliwości rozwiązania tego problemu, takie jak mało efektywna metoda przeglądania zupełnego (bruteforce, metoda siłowa), czy algorytm aproksymacyjny, który w zachłannej wersji sortuje elementy w kolejności malejącej porównując stosunek wartości do wagi przedmiotu, następnie wstawia je kolejno do plecaka, gdy nie mieści się, jest pomijany. W ten sposób wybierany jest maksymalny wynik upakowania plecaka. Nie jest to jednak metoda optymalna. Znacznie lepszym sposobem jest metoda programowania dynamicznego, którą zaimplementowałem w programie.

Programowanie dynamiczne pozwala na znalezienie optymalnego rozwiązania problemu plecakowego. Wykorzystuje metodę "dziel i zwyciężaj", dzieląc zagadnienie na szereg mniejszych problemów. Dlatego rozpoczyna się od sytuacji, w której plecak należałoby wypełnić tylko jednym rodzajem przedmiotów. Dodatkowo pojemność maksymalna plecaka również zostaje podzielona i rozszerzana co jednostkę od 1 do wartości maksymalnej. W ten sposób stworzyć można tabelę P upakowania plecaka. Każdy kolejny wiersz takiej tablicy bazuje na poprzedzającym go wierszu, a dopisywane kolejne przedmioty możliwe do włożenia do plecaka są porównywane pod względem optymalności, czy lepszym rozwiązaniem będzie użycie tych nowych przedmiotów, czy

wcześniejsze już są optymalne. Dokładny opis algorytmu znaleźć można na stronie: <http://www-users.mat.uni.torun.pl/~henkej/knapsack.pdf>, na której wzorowałem się rozwiązując zadany problem.

Aby przetestować działanie algorytmu dynamicznego stworzyłem dwie sytuacje: najbardziej klasyczna, czyli problem upakowania łupów złodzieja, tak aby zmieścił jak najwięcej warte przedmioty, oraz problem spakowania najbardziej potrzebnych przedmiotów na biwak. Oba problemy zostają rozwiązane w sposób optymalny. Rozwiązanie można zobaczyć uruchamiając program "plecak".