

Helena Masłowska gr. 5.1
indeks: 148182
2 sem. Informatyka stac.

Dyskretny problem plecakowy

Link do repozytorium: <https://github.com/HelenaMaslowska/Problem-plecakowy>

Spis treści

1. Wstęp
2. Dyskretny problem plecakowy, metody rozwiązywania:
 - a. Siłowy (Brute-force)
 - b. Rekurencyjny
 - c. Dynamiczny
3. Wykresy porównujące 3 metody dla rozwiązania problemu plecakowego
4. Wnioski

Wstęp

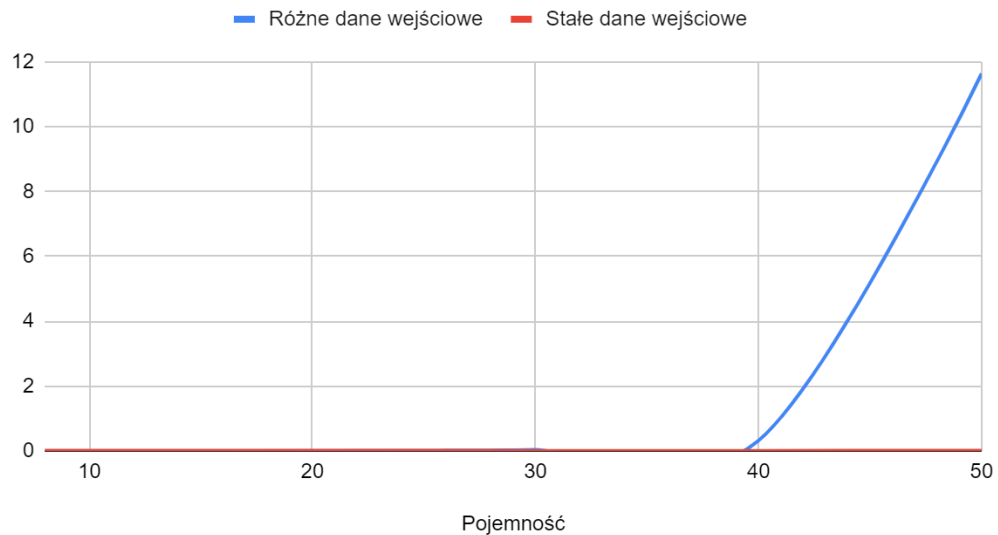
Celem niniejszego sprawozdania jest analiza 3 rozwiązań dyskretnego problemu plecakowego, metoda: siłowa, rekurencyjna bez memoizacji oraz dynamiczna. Przedstawione rozwiązania będą bazowały na różnych instancjach wejściowych. Pojemność plecaka wynosi maksymalnie 50 albo 100, w zależności od algorytmu (metoda siłowa dla plecaka, którego pojemność plecaka wynosi 100 zajmuje bardzo dużo czasu, ze względu na wykładniczą złożoność algorytmu).

Metoda siłowa - Brute force

Opis metody

Algorytm opiera się na metodzie brute force - sprawdza każde możliwe rozwiązanie problemu za pomocą binarnej reprezentacji każdej liczby od 1 do 2^n , gdzie n jest liczbą elementów w domu, wkładanych do plecaka, stąd też złożoność algorytmu $O(2^n)$.

Metoda siłowa



Spostrzeżenia i wnioski

Można zauważyć różnicę między różnymi danymi wejściowymi a stałymi danymi wejściowymi i wywnioskować, że szybkość algorytmu zależy od wielkości danych wejściowych.

Metoda rekurencyjna bez memoizacji

Opis metody

Algorytm przeszukuje możliwe rozwiązania poprzez rekurencyjne dodawanie elementu na listę, wywołując funkcję również dla opcji bez tego elementu i sprawdza, która z tych opcji jest przyniesie większą wartość w wyniku końcowym. Co istotne, algorytm nie zapisuje dotychczasowych wyników, więc musi liczyć wszystkie możliwości od nowa.

Metoda rekurencyjna



Spostrzeżenia i wnioski

Dla różnych danych wejściowych metoda rekurencyjna działa podobnie do metody siłowej, tu złożoność obliczeniowa wynosi $O(2^n)$ czyli podobnie do metody siłowej jednak warto zwrócić uwagę na to, że ilość elementów potrzebna do zauważenia jakiegokolwiek zmiany w czasie działania wyniku musiała zostać nieznacznie zwiększona. Wynika to ilości obliczeń potrzebnej do uzyskania odpowiedzi na problem.

Stałe dane wejściowe nie wydłużają czasu trwania algorytmu, gdyż rekurencja wykona się za każdym razem podobną ilość razy.

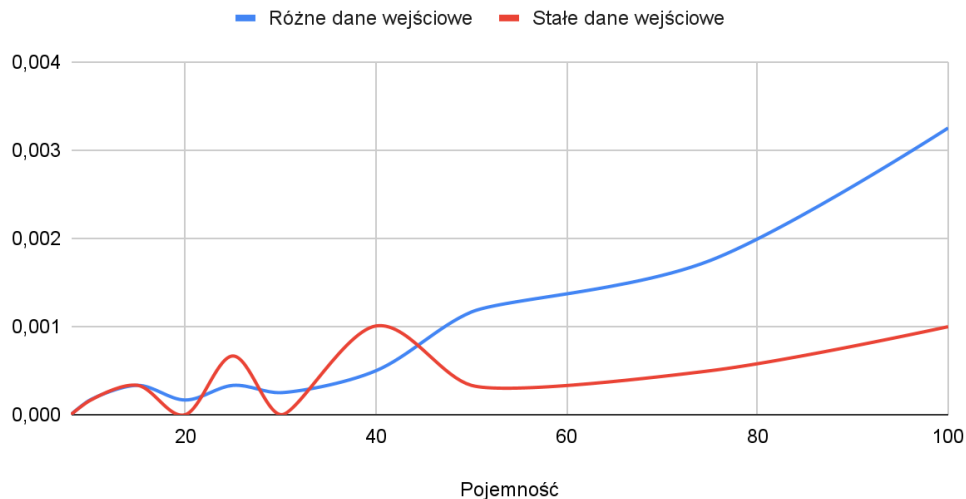
Metoda dynamiczna

Opis metody

Algorytm tworzy tablicę dwuwymiarową, która liczy w kolejnych krokach aktualną wartość optymalną po dodaniu każdego elementu. Wiersze odpowiadają elementom, które można dodać do plecaka, a kolumny - pojemność plecaka. Dla każdej komórki w tablicy jest przypisywana po kolei najbardziej optymalna wartość spośród dwóch możliwych: pierwsza, jest to dodanie elementu który odpowiada aktualnemu rzędowni do plecaka (suma wartości elementu z aktualnego rzędu i sumy wartości już istniejących elementów w plecaku) oraz druga, jest to wartość z poprzedniego rzędu i tej samej kolumny (czyli tej samej pojemności plecaka, wybranie tej opcji oznaczałoby, że dodanie nowego elementu z aktualnego rzędu do plecaka nie optymalizuje wyniku).

Wartość maksymalną, jaką plecak może unieść można odczytać w ostatniej kolumnie i ostatnim rzędzie w tablicy.

Metoda dynamiczna

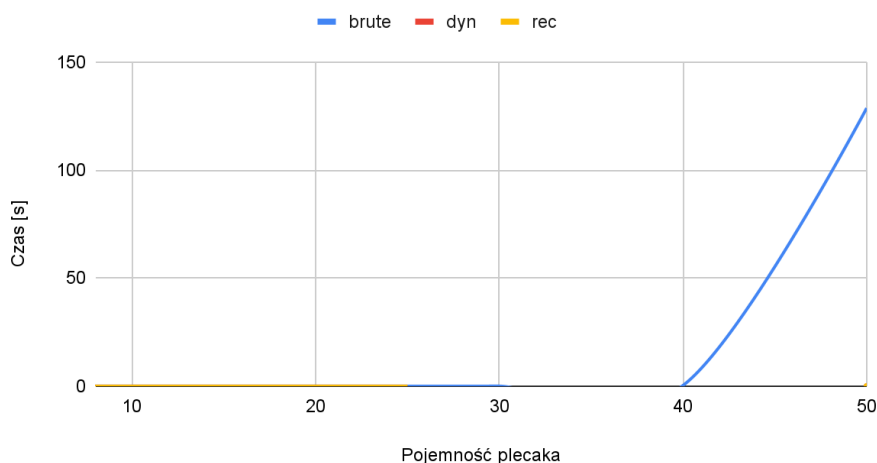


Spostrzeżenia i wnioski

Algorytm jest bardzo szybki, wykonuje się najszybciej w porównaniu do pozostałych algorytmów, odznacza się złożonością pseudowielomianową $O(n \cdot c)$, gdzie n i c to odpowiednio wielkość instancji danych wejściowych oraz pojemność plecaka. Złożoność pamięciowa również jest tyle równa $O(n \cdot c)$. Stałe dane wejściowe nieznacznie zmniejszają czas wypełnienia tablicy, tym samym ukończenia algorytmu.

Wykresy porównujące 3 metody dla rozwiązania problemu plecakowego

Proporcjonalne dane wejściowe dla różnej pojemności plecaka

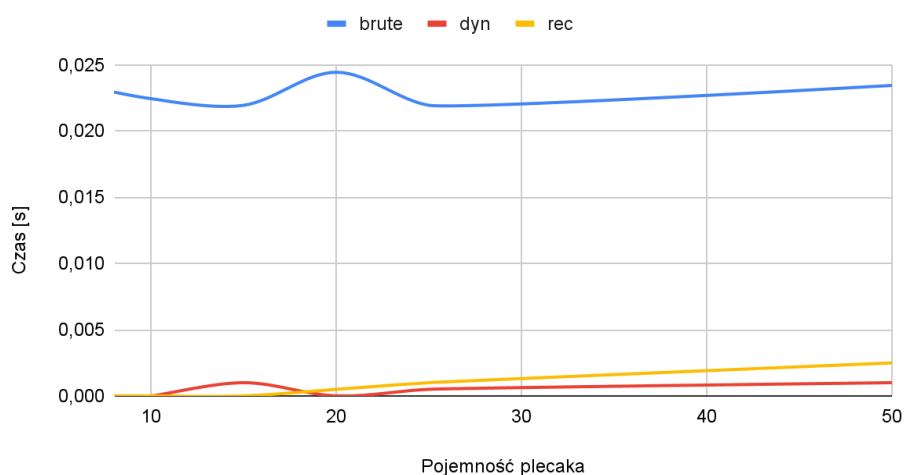


Spostrzeżenia

Linie wykresu dynamicznego i rekurencyjnego rozwiązania są niewidoczne i nakładają się z linią poziomą wykresu.

Metoda siłowa jest wyraźnie najmniej efektywna, złożoność algorytmu jest wykładnicza. Pozostałe algorytmy rozwiązują problem plecakowy w dość korzystnym czasie.

Stałe dane wejściowe dla różnej pojemności plecaka



Spostrzeżenia

Pomijając niedokładności pomiarowe można zauważyć że stałe dane znacznie zmniejszają czas działania algorytmu siłowego. Algorytm dynamiczny i rekurencyjny pozostają równie szybkie.

Wnioski

W zależności od zapotrzebowania, można zastosować poszczególne metody rozwiązywania problemu plecakowego. Najbardziej optymalne jest rozwiązanie dynamiczne, jednak zajmuje ono większą pamięć w porównaniu do pozostałych algorytmów. Najprostszą w implementacji jest metoda siłowa, po rozbudowaniu algorytmu, jest możliwe wydobycie konkretnej tablicy elementów, które będą w plecaku, jednak największą jego wadą jest wykładnicza złożoność obliczeniowa. Szybszą metodą jest metoda rekurencyjna, jednak nie zapisuje ona poprzednich wyników, co znacznie wydłuża czas działania algorytmu.