

Mamy problem plecakowy.

Zadania:

1. Znaleźć rozwiązanie optymalne przez przegląd wyczerpujący.
2. Rozwiązać problem przy użyciu heurystyki: do plecaka pakujemy przedmioty według kolejności wynikającej ze stosunku p/w. Uwaga: heurystyka to nie funkcja heurystyczna. Nie używamy tu jeszcze funkcji heurystycznej i algorytmu A*.

Pytania (odpowiedzi proszę umieścić w pliku tekstowym):

- Jakie rozwiązania i jaką wartość funkcji oceny uzyskano? Czy uzyskano takie same rozwiązania?
- Jak dużą instancję problemu (liczba przedmiotów) da się rozwiązać w około minutę metodą zachłanną?
- Jak bardzo wydłuży obliczenia dodanie jeszcze jednego przedmiotu?
- Jakie wnioski można wyciągnąć na podstawie wyników tego ćwiczenia?

W pliku main.py znajduje się cały kod do tego zadania.

Do rozwiązania zadania przez przegląd wyczerpujący służy funkcja `bruteForceAlgorithm`, która korzysta z pomocniczej funkcji `allCombinations` do wyznaczenia wszystkich kombinacji przedmiotów.

Do rozwiązania zadania przy użyciu heurystyki służy funkcja `heuristicAlgorithm`, która korzysta z pomocniczej funkcji `itemRatio` do wyznaczania współczynnika wartości przedmiotu do jego wagi

Odpowiedzi na pytania:

- W przypadku pierwszego rozwiązania otrzymujemy rozwiązanie optymalne, natomiast w przypadku heurystyki możemy dostać to samo rozwiązanie lub gorsze w zależności od danych wejściowych
- W moim przypadku w minutę mogłem rozwiązać ten problem maksymalnie dla 24 przedmiotów
- Dodanie kolejnego przedmiotu wydłużyło czas oczekiwania prawie 4-krotnie (z 40 sekund do 2 min 30 sek)
- Zwykle pierwsze rozwiązanie jakie przyjdzie nam do głowy jest mało opłacalne i warto poszukać lepszego algorytmu