**WSI – Ćwiczenia 1 – Zadanie 1**

*Mateusz Kołacz, 336360*

Mamy problem plecakowy jak na wykładzie:

m = np.array([8, 3, 5, 2]) #masa przedmiotów

M = np.sum(m)/2 #niech maksymalna masa plecaka będzie równa połowie masy przedmiotów

p = np.array([16, 8, 9, 6]) #wartość przedmiotów

**Zadania:**

1. Znaleźć rozwiązanie optymalne przez przegląd wyczerpujący (analizuje wszystkie kombinacje).

2. Rozwiązać problem przy użyciu heurystyki: do plecaka pakujemy przedmioty według kolejności wynikającej ze stosunku p/m.

Przeglądamy listę przedmiotów do końca, chyba że plecak jest już pełen lub zostało w nim tak mało miejsca, że już na pewno nic się nie zmieści. 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 przeglądu wyczerpującego?
* Ile czasu zajmie rozwiązanie tego problemu metodą zachłanną (używając heurystyki)? Odpowiednio długie wektory m i p należy wylosować, M = np.sum(m)/2.
* Jak bardzo wydłuży obliczenia dodanie jeszcze jednego przedmiotu?
* Jakie wnioski można wyciągnąć na podstawie wyników tego ćwiczenia?

**Uwagi:**

*\* Nie używać rekurencji.*

*\* Kod którego czas wykonania mierzymy nie powinien wypisywać niczego na konsolę.*

*\* Nie pytałem o teoretyczną złożoność obliczeniową. Aby odpowiedzieć na pytania należy wykonać stosowne eksperymenty.*

*\* Czas działania programu można zgrubnie zmierzyć komendą "time", po której podaje się nazwę badanego programu. Zbadanie konkretnego fragmentu kodu lepiej jednak zrobić w Pythonie, np. tak:*

import time

start = time.process\_time()

#tu badany kod

end = time.process\_time()

total = end - start

print("{0:02f}s".format(total))

**Odpowiedzi:**

**Pyt. 1: Jakie rozwiązania i jaką wartość funkcji oceny uzyskano? Czy uzyskano takie same rozwiązania?**

**Odp.:** Dla przeglądu wyczerpującego otrzymano rozwiązanie optymalne **0110** o wartości **17** i wadze **8**, natomiast przy użyciu heurystyki otrzymano rozwiązanie suboptymalne **0101** o wartości **14** i wadze **5**.

Nie, nie uzyskano takich samych rozwiązań.

**Pyt. 2: Jak dużą instancję problemu (liczba przedmiotów) da się rozwiązać w około minutę metodą przeglądu wyczerpującego? Ile czasu zajmie rozwiązanie tego problemu metodą zachłanną (używając heurystyki)? Odpowiednio długie wektory m i p należy wylosować, M = np.sum(m)/2.**

**Odp.:** Przy wykorzystaniu metody przeglądu wyczerpującego możliwe jest rozwiązanie w około minutę instancje problemu o wielkości 23 przedmiotów.

Rozwiązanie tego problemu metoda zachłanna dla instancji 23 przedmiotów zajmie średnio **0,0008** sekundy.

**Pyt. 3: Jak bardzo wydłuży obliczenia dodanie jeszcze jednego przedmiotu?**

**Odp.:** Dodanie jeszcze jednego przedmiotu, np.: z 23 do 24 wydłuży obliczenia średnio o ok. 100% sekund (2-krotnie).

ALGORYTM WYCZERPUJĄCY:

Parametr min. średnia od. stand. maks.

0 mu=5 0,000746 0,000784 0,000030 0,000893

1 mu=10 0,004083 0,004253 0,000268 0,005181

2 mu=15 0,134719 0,137423 0,003380 0,148592

3 mu=20 5,160117 5,545770 0,140600 5,675161

4 mu=23 45,792929 48,508502 1,352291 50,793413

5 mu=24 94,063071 94,784682 0,692498 96,676439

AGORYTM ZACHŁANNY:

Parametr min. średnia od. stand. maks.

0 mu=5 0,000901 0,000945 0,000036 0,001094

1 mu=10 0,000741 0,000853 0,000054 0,000938

2 mu=15 0,000743 0,000755 0,000008 0,000777

3 mu=20 0,000746 0,000756 0,000006 0,000770

4 mu=23 0,000752 0,000829 0,000115 0,001175

5 mu=24 0,000752 0,000762 0,000008 0,000781

**Pyt. 4: Jakie wnioski można wyciągnąć na podstawie wyników tego ćwiczenia?**

**Odp.:** Użycie przeglądu wyczerpującego jest wolniejsze niż użycie algorytmu zachłannego. Jednakże, przegląd wyczerpujący zawsze zwraca rozwiązanie optymalne. Algorytm zachłanny jest natomiast o wiele szybszy, ale w najgorszym przypadku zwraca on rozwiązanie suboptymalne i nie mamy gwarancji, ze rozwiązanie będzie optymalne globalnie.