

### **Część programistyczna**

Wyobraź sobie, że przygotowujesz konfigurację wyposażenia dla ekspedycji naukowej prowadzącej badania terenowe w trudno dostępnym regionie. Do dyspozycji zespołu jest ograniczona przestrzeń ładunkowa w samochodzie terenowym, stanowiącym jedyny środek transportu podczas wyprawy. Ograniczenie to determinuje, jaką ilość sprzętu można zabrać w ramach jednej misji. Każdy element wyposażenia charakteryzuje się dwoma parametrami: zajmowaną przestrzenią (rozmiarem, wagą lub objętością) oraz użytecznością operacyjną, rozumianą jako wkład w realizację celów badawczych i technicznych ekspedycji (np. zbieranie danych, zapewnienie bezpieczeństwa, komunikacja, analiza próbek). Twoim zadaniem jest wybór takiego zestawu wyposażenia, aby nie przekroczyć dostępnej pojemności ładunkowej, a jednocześnie zmaksymalizować całkowitą użyteczność wybranego sprzętu.

Należy zaimplementować trzy algorytmy, które pozwolą w sposób automatyczny skompletować wyposażenie ekspedycji w warunkach ograniczonej przestrzeni ładunkowej:

1. Algorytm programowania dynamicznego (AD),
2. Algorytm zachłanny (AZ) wykorzystujący współczynnik opłacalności (rozumiany jako użyteczność operacyjna na jednostkę zajmowanej przestrzeni),
3. Algorytm siłowy (AB) przeglądający wszystkie dopuszczalne kombinacje wyposażenia kodujący rozwiązania w postaci liczb w zapisie binarnym.

Program powinien:

1. Generować losowe dane wejściowe oraz wczytywać dane z pliku tekstowego, w którym w pierwszej linii podane są liczba dostępnych elementów wyposażenia ( $n$ ) oraz dostępna pojemność ładunkowa ( $b$ ), a w kolejnych liniach: rozmiary i użyteczności operacyjne tych elementów
2. Być odpornym na błędy w danych wejściowych.
3. Dla każdego uruchomienia podawać: czas obliczeń, wartość uzyskanego rozwiązania (łącznie użyteczność operacyjną), sumaryczny rozmiar wybranego zestawu elementów, identyfikatory elementów wchodzących w skład wybranego wyposażenia.
4. W trybie demonstracyjnym (dla niewielkich instancji, np.  $n \leq 15$ ) program powinien dodatkowo prezentować: kolejne decyzje podejmowane przez algorytm zachłanny, tabelę programowania dynamicznego dla algorytmu AD, sygnatury podproblemów mających więcej niż jedno rozwiązanie (sygnatura w postaci: liczba elementów-pojemność), liczbę sprawdzonych konfiguracji w algorytmie siłowym (w tym liczbę rozwiązań dopuszczalnych oraz liczbę rozwiązań optymalnych).
5. Dla algorytmu zachłannego należy dodatkowo określić, czy uzyskane rozwiązanie jest optymalne, poprzez porównanie z wynikiem algorytmu dokładnego (AD lub AB).

### **Część eksperymentalna**

Generowanie danych: należy wygenerować zbiory przedmiotów o liczności  $n$ . Każdy przedmiot powinien posiadać: identyfikator ( $i$ ), rozmiar  $r(i)$ , użyteczność operacyjną  $u(i)$ , przy czym rozmiary i użyteczności powinny być liczbami naturalnymi. Pojemność miejsca ładunkowego w samochodzie terenowym  $b$  należy dobierać eksperymentalnie tak, aby możliwe było zaobserwowanie różnic pomiędzy algorytmami.

Trzy niezależne eksperymenty:

1. Eksperyment 1 – analiza wpływu liczby przedmiotów na efektywność algorytmów. Należy wybrać 10–15 wartości  $n$  rosnących liniowo oraz stałą pojemność miejsca ładunkowego. Dla każdej wartości  $n$  należy: (i) wygenerować 10 instancji problemu, (ii) uruchomić wszystkie algorytmy, (iii) zmierzyć ich czasy działania, (iv) obliczyć średni czas działania każdego algorytmu i odchylenie standardowe.
2. Eksperyment 2 – analiza wpływu pojemności miejsca ładunkowego na efektywność algorytmów. Należy wybrać stałą liczbę przedmiotów  $n$ . Następnie przeprowadzić pomiary dla 10–15 różnych wartości pojemności miejsca ładunkowego  $b$  rosnących liniowo. Dla każdej wartości  $b$  należy: (i) wygenerować 10 instancji problemu, (ii) uruchomić wszystkie algorytmy, (iii) obliczyć średni czas działania każdego algorytmu oraz odchylenie standardowe.

## Zadanie 5: Algorytmy dokładne i heurystyczne dla problemu plecakowego

---

3. Eksperyment 3 – jakość algorytmu zachłannego. Należy wygenerować dużą liczbę losowych instancji problemu (co najmniej 1000), następnie dla każdej instancji: (i) wyznaczyć rozwiązanie optymalne (AD), (ii) wyznaczyć rozwiązanie algorytmem zachłannym, (iii) porównać uzyskane wyniki. Następnie należy określić: procent przypadków, w których algorytm zachłanny znalazł rozwiązanie optymalne, średnią jakość rozwiązania zachłannego względem optimum, największe zaobserwowane odchylenie od optimum.

Wyniki pomiaru czasu uzyskane we wszystkich eksperymentach należy zapisać w tabelach.

### **Sprawozdanie**

Sprawozdanie powinno zawierać:

1. Wykres liniowy przedstawiający zależność czasu obliczeń  $t$  (oś Y) od liczby elementów wyposażenia  $n$  (oś X) przy stałej pojemności miejsca ładunkowego  $b$ . Na wykresie należy umieścić 3 krzywe – po jednej dla każdego algorytmu. Dla każdego punktu pomiarowego należy zaznaczyć odchylenie standardowe w postaci słupka błędów.
2. Wykres liniowy przedstawiający zależność czasu obliczeń  $t$  (oś Y) od pojemności miejsca ładunkowego  $b$  (oś X) przy stałej liczbie elementów  $n$ . Na wykresie należy umieścić 3 krzywe odpowiadające wszystkim algorytmom. Dla każdego punktu pomiarowego należy zaznaczyć odchylenie standardowe w postaci słupka błędów.
3. Trzy wykresy powierzchniowe 3D  $t=f(n,b)$  zależności czasu obliczeń  $t$  od liczby  $n$  elementów wyposażenia i pojemności miejsca ładunkowego  $b$  – oddzielny wykres dla każdego algorytmu.
4. Wykres jakości rozwiązań algorytmu zachłannego. Dla wszystkich instancji z eksperymentu 3 należy obliczyć względny błąd rozwiązania:  $e = 100((f^* - f_s)/f^*)$ , gdzie  $f^*$  to optymalna wartość funkcji celu, a  $f_s$  – wartość rozwiązania suboptymalnego znalezione przez algorytm zachłanny. Następnie należy przedstawić histogram rozkładu błędów (przykładowe zakresy na osi X: 0, 0-5, 5-10, 10-20, >20), na osi Y liczba instancji). Na podstawie wykresu należy ocenić, jak często algorytm zachłanny znajduje rozwiązanie optymalne oraz jak duże są typowe odchylenia od optimum.
5. Analizę wyników eksperymentów obliczeniowych. Należy przeanalizować wyniki eksperymentów dla wszystkich algorytmów i odpowiedzieć na pytania:
  - Jak rośnie czas działania algorytmu siłowego wraz ze wzrostem liczby elementów i dla jakich wartości  $n$  czas ten staje się nieakceptowalnie długi z perspektywy praktycznej?
  - Czy pojemność miejsca ładunkowego  $b$  ma zauważalny wpływ na efektywność algorytmu siłowego?
  - Czy algorytm programowania dynamicznego zawsze jest szybszy od algorytmu siłowego?
  - Który z zaimplementowanych algorytmów dokładnych jest bardziej wrażliwy na zmianę liczby przedmiotów  $n$ , a który na zmianę dostępnej pojemności  $b$ ?
  - Jak często algorytm zachłanny znajduje rozwiązanie optymalne?
  - Jak bardzo rozwiązania zachłanne odbiegają od optimum?
  - Czy można wskazać charakterystyczne przypadki, dla których algorytm zachłanny zawodzi?
6. Analizę złożoności obliczeniowej. Dla każdego algorytmu należy podać złożoność czasową i pamięciową, porównać wyniki teoretyczne z wynikami eksperymentalnymi, wskazać przyczyny ewentualnych rozbieżności. Ponadto należy określić, do jakich klas złożoności należy optymalizacyjna oraz decyzyjna wersja problemu plecakowego.
7. Wnioski końcowe. W podsumowaniu należy wskazać najbardziej efektywny algorytm w praktyce, określić, kiedy warto stosować programowanie dynamiczne, ocenić użyteczność algorytmu zachłannego, wskazać ograniczenia algorytmu siłowego, porównać kompromis pomiędzy szybkością działania a jakością rozwiązania.