2-SAT (P/NP)

Polega na przypisywaniu wartości zmiennym boolowskim, z których każda ma dwie możliwe wartości, w celu spełnienia systemu ograniczeń na parach zmiennych.
Dwa literały w klauzuli CNF.

$$(x_0 \vee x_2) \wedge (x_0 \vee \neg x_3) \wedge (x_1 \vee \neg x_3) \wedge (x_1 \vee \neg x_4) \wedge (x_2 \vee \neg x_4) \wedge (x_0 \vee \neg x_5) \wedge (x_1 \vee \neg x_5) \wedge (x_2 \vee \neg x_5) \wedge (x_3 \vee x_6) \wedge (x_4 \vee x_6) \wedge (x_5 \vee x_6).$$

3-SAT (NP-zupełny)

Problem sprawdzenia czy zadana formuła logiczna jest spełniona. Trzy literały w klauzuli CNF.

$$\text{3SAT}\,(\quad (a \vee b \vee \neg c) \wedge (\neg a \vee \neg b \vee d) \wedge (\neg a \vee b \vee \neg d) \wedge (b \vee \neg c \vee d)\,)$$

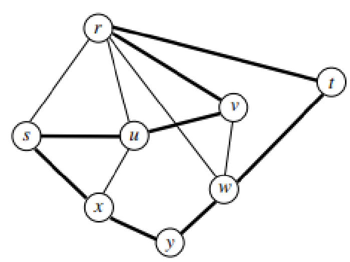
Cyklu Eulera (P/NP)

Kolokwialnie rysowanie koperty bez odrywania długopisu. Cykl Eulera, to taki cykl w grafie, która zawiera każdą krawędź grafu, przy czym każdą dokładnie raz. Natomiast łańcuch Eulera to taka droga (czyli nie musi się kończyć tam, gdzie zaczyna), która zawiera każdą krawędź grafu dokładnie raz.

Dotyczy to zarówno grafów nieskierowanych, jak i skierowanych.

Cyklu Hamiltona (NP)

Dla danego grafu G sprawdzić, czy istnieje cykl prosty zawierający wszystkie wierzchołki G. Czy można obejść wszystkie wierzchołki i wrócić do początkowego tak, aby każdy wierzchołek odwiedzić dokładnie raz?



Komiwojażera (TSP – TRAVELING SALESMAN PROBLEM) (NP)
Dane wejściowe:

- n – miast ponumerowanych od 1 do n
- c (i, j) – koszt podróży z miasta i do j
- k – liczba rzeczywista, maksymalny dopuszczalny koszt całkowity podróży

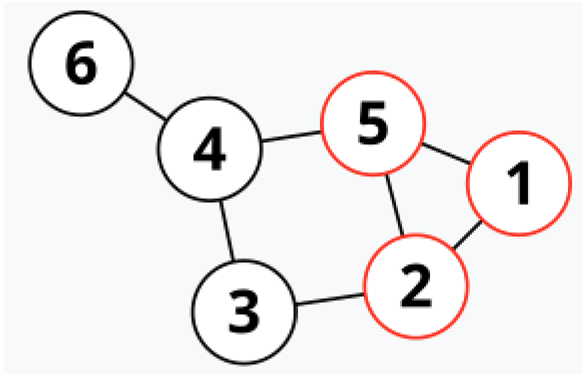
Komiwojażer mieszka w mieście 1. Chce odwiedzić wszystkie miasta i wrócić do miasta 1 tak, aby całkowity koszt podróży był mniejszy lub równy k, czy jest to możliwe?

Klika (NP)

Dane wejściowe:

- G – graf
- k – dodatnia liczba całkowita

Modelując każdego osobnika w postaci wierzchołka, a powiązania między osobnikami jako nieskierowane krawędzie.
Klika reprezentuje grupę osobników, wśród których każdy ma połączenie z każdym.
Pytamy, czy G ma klikę o rozmarze k?



Klika rozmiaru 3, zaznaczona na czerwono.

Pokrycia
Pokrycia wierzchołkowego (VERTEX COVER) (NP)

Dane wejściowe:

- G – graf
- m – dodatnia liczba całkowita

Rozmiar pokrycia wierzchołkowego jest liczbą zawieranych przez nie wierzchołków.
Pytamy, czy G ma pokrycie wierzchołkowe rozmiaru m.
Pokrycie wierzchołkowe grafu nieskierowanego G jest takim podzbiorem S wierzchołków, że każda krawędź G jest incydentna z co najmniej jednym wierzchołkiem S. Mówimy, że każdy wierzchołek S "pokrywa" incydentne z nim krawędzie.

Plecakowy (KNAPSACK PROBLEM) (NP)

Dane wejściowe:

- n – dany zbiór przedmiotów o znanej wadze W i znanej wartości V
- W – ograniczenie wagi
- V – ograniczenie wartości

Chcemy zapakować do plecaka podzbiór najcenniejszych przedmiotów najcenniejszych przedmiotów nie przekraczając limitu wagowego.

Najkrótszej ścieżki z jednym źródłem (SINGLE-SOURCE SHORTEST PATH) (P/NP)
Algorytmy do rozwiązania: Algorytm Dijkstry, Algorytm Bellmana-Forda

Dane wejściowe:

- G – graf
- w – wagi krawędzi
- s – wybrany wierzchołek źródłowy

Skupia się na znalezieniu najkrótszych ścieżek od jednego wybranego wierzchołka źródłowego do wszystkich innych wierzchołków.

Najkrótszej ścieżki pomiędzy parami wszystkich wierzchołków (P/NP)
Algorytm do rozwiązania: Algorytm Floyda-Warshalla, Algorytm Johnsona

Wymaga obliczeń dla każdej pary wierzchołków, co jest bardziej złożone i kosztowne obliczeniowo, szczególnie dla dużych grafów.

Dane wejściowe:

- G – graf
- w – wagi krawędzi

Twierdzenie o rekurencji:

Niech $a \geq 1$ i $b > 1$ będą stałymi i $f(n)$ będzie pewną funkcją i niech $T(n)$ będzie zdefiniowane dla nieujemnych liczb całkowitych przez rekurencję

$$T(n) = aT(\frac{n}{b}) + f(n).$$

Wtedy funkcja $T(n)$ może być ograniczona asymptotycznie w następujący sposób

1. jeżeli $f(n) = O(n^{\log_b a - \epsilon})$ dla $\epsilon > 0$, to $T(n) = \Theta(n^{\log_b a})$,
2. jeżeli $f(n) = \Theta(n^{\log_b a})$, to $T(n) = \Theta(n^{\log_b a} \lg n)$,
3. jeżeli $f(n) = \Omega(n^{\log_b a + \epsilon})$ dla $\epsilon > 0$ i $af(\frac{n}{b}) \leq cf(n)$ dla pewnej stałej $c < 1$, to $T(n) = \Theta(f(n))$.

Twierdzenie o rekurencji uniwersalnej opisuje czas działania algorytmu, który dzieli problem rozmiaru n na a podproblemów, każdy rozmiaru $\frac{n}{b}$. Każdy z a podproblemów jest rozwiązywany rekurencyjnie w czasie $T(\frac{n}{b})$. Koszt dzielenia problemów oraz łączenia rezultatów częściowych jest opisany funkcją $f(n)$.