**Algoritmul lui Kruskal**

**Algoritmul lui Kruskal** este un [algoritm](https://ro.wikipedia.org/wiki/Algoritm) în [teoria grafurilor](https://ro.wikipedia.org/wiki/Teoria_grafurilor) care găsește [arborele](https://ro.wikipedia.org/wiki/Arbore_(informatic%C4%83)) parțial de cost minim pentru un [graf](https://ro.wikipedia.org/wiki/Graf) [conex](https://ro.wikipedia.org/wiki/Conexitate) ponderat. Cu alte cuvinte, găsește submulțimea muchiilor care formează un arbore care include toate vârfurile și care este minimizat din punct de vedere al costului. Dacă graful nu este conex, atunci algoritmul găsește *un arbore parțial de cost minim pentru fiecare componentă conexă*. Algoritmul lui Kruskal este un exemplu de [algoritm greedy](https://ro.wikipedia.org/wiki/Algoritm_greedy).

Algoritmul funcționează în felul următor:

* creează un arbore *F* (o mulțime de arbori), unde fiecare vârf din graf este un arbore separat
* creează o mulțime *S* care conține toate muchiile din graf
* atât timp cât *S* este nevidă
  + elimină o muchie de cost minim din *S*
  + dacă acea muchie conectează doi arbori distincți, atunci adaugă muchia în arbore, combinând cei doi arbori într-unul singur
  + altfel, ignoră muchia

La sfârșitul algoritmului, arborele are doar o componentă care reprezintă un arbore parțial de cost minim al grafului.

## **Exemplu**



# Algoritmul lui Prim

**Algoritmul lui Prim** este un algoritm din [teoria grafurilor](https://ro.wikipedia.org/wiki/Teoria_grafurilor) care găsește arborele parțial de cost minim al unui graf conex ponderat. Înseamnă că găsește submulțimea muchiilor care formează un arbore care include toate vârfurile și al cărui cost este minimizat.

**Exemplu**

