ARBORI ȘI PĂDURI

UNIVERSITATEA TEHNICĂ DIN CLUJ-NAPOCA

5. ARBORI SI PADURI

Definitii:

- nr. ciclomatic
- nr. cociclomatic
- arbore
- padure
- arbore cu radacina sau arborescenta
- arbore de acoperire
- problema det. arbore de acoperire val. Min./max. graf G conex dat.

Proprietati. Teoreme

- nr. ciclomatic, cociclomatic: nr. nenegative
- Numar ciclomatic 0
- Numar ciclomatic 1
- nr. ciclomatic = nr. cicluri elementare
- Obs.: Alg. Kruskal

- Numar ciclomatic si numar cociclomatic
- Arbori si paduri
- Arbori
- Arbori cu radacina
- Arbori de acoperire
- Arbori de acoperire de valoare optima

Algoritmi (descriere+pseudocod+ex.)

det. Arbori de acoperire de valoare optima

- alg. **Kruskal**
- alg. **Prim**

Proprietati. Teoreme

- T.graf neorientat, adaugarea muchiei 2 noduri neadiacente=> exact una din afirmatii are loc (1,2)
- 5 Consecinte teorema
- T. caracterizare a arborilor (1-6) (+demonstratie)

Număr ciclomatic și număr cociclomatic

Fie $G = (X, U)$ un graf orientat de ordin n, dimensiune m și cu p componente conexe.

Definiție

Numărul $\rho(G) = m - n + p$ se numește **numărul ciclomatic** al grafului G.

Definiție

Numărul $\nu(G) = n - p$ se numește **numărul cociclomatic** al grafului G.

$$\rho(G) = m - n + p = m - \nu(G)$$

Număr ciclomatic și număr cociclomatic

Teoremă

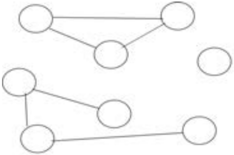
Fie $G = (X, U)$ un graf neorientat, iar $G' = (X, U')$ graful obținut din G prin adăugarea muchiei (i, j) între două noduri neadiacente în G, $(i, j) \notin U$, $U' = U \cup \{(i, j)\}$.
Atunci exact una dintre următoarele afirmații are loc:
1. $\rho(G') = \rho(G)$ și $\nu(G') = \nu(G) + 1$
2. $\rho(G') = \rho(G) + 1$ și $\nu(G') = \nu(G)$.

- $n' = n$, $m' = m + 1$, p' nr. noduri, muchii, componente conexe în G'

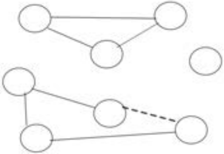
1. nodurile i și j în aceeași componentă conexă $\Rightarrow p' = p$.
- $\rho(G') = m' - n' + p' = m + 1 - n + p = \rho(G) + 1$
$\nu(G') = n' - p' = n - p = \nu(G)$

2. nodurile i și j nu în aceeași componentă conexă,
$i \in C_1$, $j \in C_2 \Rightarrow p' = p - 1$
- $\rho(G') = m' - n' + p' = m + 1 - n + p - 1 = m - n + p = \rho(G)$,
- $\nu(G') = n' - p' = n - p + 1 = \nu(G) + 1$.

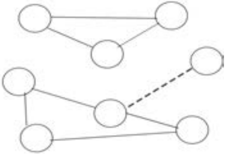
Număr ciclomatic și număr cociclomatic



n=8, m=6, p=3
ρ=1,ν=5



n=8, m=7, p=3
Cazul 1, ρ1=2,ν1=5



n=8, m=8, p=2
Cazul 2, ρ2=2,ν2=6

Proprietati si consecinte teorema:

- oricare ar fi graful $G = (X, U)$, numărul ciclomatic și numărul cociclomatic sunt numere nenegative, $\rho(G) \geq 0, \ \nu(G) \geq 0$

- $G = (X, U)$ se obține adăugând muchii grafului $G_0 = (X, \emptyset)$

- în G_0, $n_0 = n, \ p_0 = n, \ m_0 = 0$

- $\rho(G_0) = m_0 - n_0 + p_0 = 0, \ \nu(G_0) = n_0 - p_0 = 0$

- conform teoremei, la fiecare muchie adăugată numerele ciclomatic și cociclomatic cresc sau rămân constante, deci în final acestea nu vor fi negative

Număr ciclomatic și număr cociclomatic

Numar ciclomatic nul

- într-un graf $G = (X, U)$ numărul ciclomatic este $\rho(G) = 0$ dacă și numai dacă G nu conține cicluri

- $\rho(G) = 0$ înseamnă că la fiecare adăgare de muchie pornind de la G_0, ne-am aflat în cazul 2 al teoremei, deci am adăugat muchii doar între componente conexe diferite

- nu s-a format nici un ciclu.

Numar ciclomatic 1

- într-un graf $G = (X, U)$ numărul ciclomatic este $\rho(G) = 1$ dacă și numai dacă G conține un singur ciclu

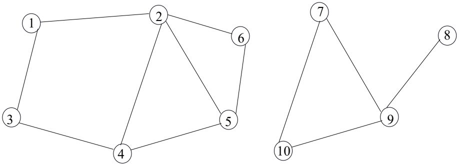
- $\rho(G) = 1$ înseamnă ne-am aflat în cazul 1 al teoremei o singură dată, deci am adăugat muchie în aceeași componentă conexă o singură dată

- s-a format un singur ciclu

- un graf care nu conține cicluri se numește graf **aciclic**.

Număr ciclomatic și număr cociclomatic

- numărul ciclomatic al unui graf reprezintă numărul ciclurilor elementare ale unui graf



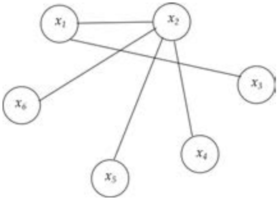
$\rho(G) = m - n + p = 12 - 10 + 2 = 4$ cicluri elementare
$\mu_1 = (1, 2, 4, 3, 1)$, $\mu_2 = (2, 6, 5, 2)$, $\mu_3 = (2, 4, 5, 2)$, $\mu_4 = (7, 9, 10, 7)$

- cicluri neelementare- reuniune de cicluri elementare
$\mu_5 = (1, 2, 6, 5, 4, 3, 1) = \mu_1 \cup \mu_2 \cup \mu_3$,
$\mu_6 = (2, 6, 5, 4, 2) = \mu_2 \cup \mu_3, \ldots$

Definiție

Se numește **arbore** un graf $G = (X, U)$ conex și fără cicluri.

- $G = (X, U)$ cu $X = \{x_1, x_2, x_3, x_4, x_5, x_6\}$ și
  $U = \{(x_1, x_2), (x_1, x_4), (x_2, x_4), (x_2, x_5), (x_2, x_6)\}$
- G are $n = 6$ noduri, $m = 5$ muchii și este arbore



Arbori și păduri

Teoremă de caracterizare a arborilor

Fie $G = (X, U)$ un graf de ordin n. Afirmațiile următoare sunt echivalente:

1. Graful G este arbore.
2. Graful G este fără cicluri și are $m = n - 1$ muchii.
3. Graful G este conex și are $m = n - 1$ muchii.
4. Graful G este fără cicluri și prin adăugarea unei muchii între oricare două vârfuri neadiacente se formează un ciclu.
5. Graful G este conex și eliminând orice muchie devine neconex.
6. Între oricare două noduri ale grafului există un singur lanț.

Demonstrație:

1) $\Rightarrow$ 2) $\quad G$ arbore $\Rightarrow G$ fără cicluri și $m = n - 1$

G este arbore $\Rightarrow$ conex $\Rightarrow p = 1$
$\Rightarrow$ fără cicluri $\Rightarrow \rho(G) = m - n + p = 0$

- atunci $\rho(G) = 0 \Leftrightarrow m - n + 1 = 0 \Leftrightarrow m = n - 1 \Rightarrow$
$\Rightarrow G$ este fără cicluri și cu $m = n - 1$

2) $\Rightarrow$ 3) $\quad G$ fără cicluri și $m = n - 1 \Rightarrow G$ conex cu $m = n - 1$

G este fără cicluri $\Rightarrow \rho(G) = m - n + p = 0$

- $\rho(G) = 0 \Leftrightarrow m - n + p = 0 \Leftrightarrow n - 1 - n + p \Leftrightarrow p = 1 \Rightarrow G$ conex și cu $m = n - 1$.

3) $\Rightarrow$ 4) G conex cu $m = n - 1 \Rightarrow G$ fără cicluri și $G + u$ exact un ciclu

- $\rho(G) = m - n + p = n - 1 - n + 1 = 0 \Rightarrow G$ nu are cicluri
- fie $x, y \in X,\ (x, y) \notin U$ și $G' = G + (x, y)$
- $n' = n,\ m' = m + 1 = n - 1 + 1 = n,\ p' = 1$
- $\rho(G') = m' - n' + p' = n - n + 1 = 1 \Rightarrow G'$ conține exact un ciclu

4) $\Rightarrow$ 5) G fără cicluri și $G+u$ exact un ciclu $\Rightarrow$ G conex și $G-u$ neconex

- G aciclic $\Rightarrow \rho(G) = m - n + p = 0$ și $\rho(G + (x,y)) = 1, (x,y) \notin U$
- pp. G nu este conex $\Rightarrow p \geq 2$
- fie $x \in C_1,\ y \in C_2 \Rightarrow (x,y) \notin U$
- $G' = G + (x,y),\ n' = n,\ m' = m+1,\ p' = p-1,$

  $\Rightarrow \rho(G') = m' - n' + p' = m - n + p = 0 \Rightarrow G'$ aciclic, contradicție

  $\Rightarrow G$ conex, $p = 1$
- fie $G" = G - (x', y'),\ (x', y') \in U$
- $n" = n,\ m" = m - 1$ și $\rho(G") = 0$ (graf aciclic cu muchie eliminată)

$\rho(G") = m" - n" + p" = 0 \Leftrightarrow m - 1 - n + p" = 0 \Leftrightarrow p" = p + 1 \Rightarrow p" = 2$, adică $G"$ este neconex

5) $\Rightarrow$ 6) G conex și $G-u$ neconex $\Rightarrow$ între $x, y \in X$ există unic lanț

G conex $\Rightarrow$ există lanț între oricare două noduri $x, y \in X$

- pp. există $x, y \in X$ între care există două lanțuri distincte μ_1, μ_2,
$\mu_1 = (x, x_{i_1}, x_{i_2}, ..., x_{i_k}, y)$ respectiv $\mu_2 = (x, x_{j_1}, x_{j_2}, ..., x_{j_l}, y)$
- x, y sunt pe un ciclu, $\mu = (x, x_{i_1}, x_{i_2}, ..., x_{i_k}, y, x_{j_l}, ..., x_{j_2}, x_{j_1}, x)$
- dacă se elimină o muchie u din ciclul μ, graful $G' = G - u$ rămâne conex, contradicție cu 5)

6) $\Rightarrow$ 1) între $\forall\ x, y \in X$ există unic lanț $\Rightarrow$ G arbore

- fie $x, y \in X \Rightarrow$ între x și y există lanț, deci G este conex
- pp. există ciclu în graful G, $\mu = (x, x_{i_1}, x_{i_2}, ..., x_{i_k}, x)$ și fie $x_{i_a}, x_{i_b} \in X$, $a, b \in \{1, 2, ..., k\}$ două noduri de pe ciclul μ
- între nodurile x_{i_a}, x_{i_b} există două lanțuri distincte,

$\mu_1 = (x_{i_a}, x_{i_{a+1}}, ..., x_{i_b})$ respectiv

$\mu_2 = (x_{i_b}, x_{i_{b+1}}, ..., x, x_{i_1}, ..., x_{i_a})$,

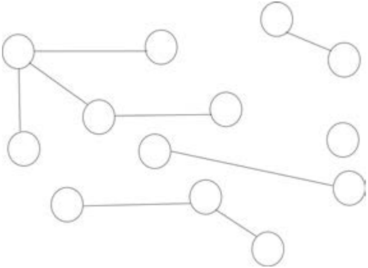
contradicție cu 6)

Arbori

Definiție

Un graf $G = (X, U)$ aciclic se numește **pădure**.

- pădurile nu sunt grafuri conexe
- fiecare componentă conexă a unei păduri este arbore



Arbori cu rădăcină

Arbore cu radacina:

- fie $T = (X, U)$ un arbore (graf conex și fără cicluri)

- un nod $r \in X$ al acestui arbore se poate alege ca nod special, numit **rădăcina arborelui**

- există un lanț unic de la rădăcina r la celelalte noduri ale grafului (conform teoremei de caracterizare a unui arbore)

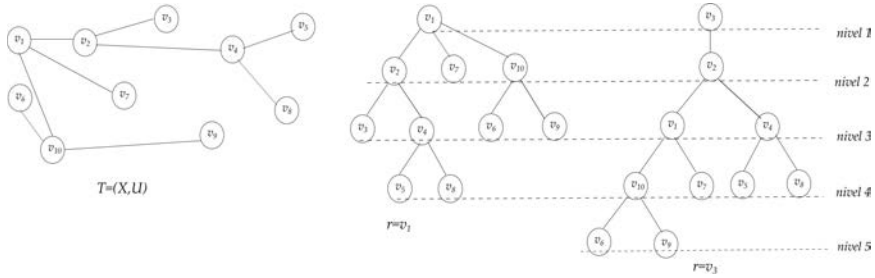
- alegerea rădăcinii duce la așezarea arborelui pe niveluri astfel:

- rădăcina este nod pe nivelul 1 (sau 0)
- pe fiecare nivel k ($k > 1$) se plasează acele vârfuri pentru care lungimea lanțurilor care le leagă de rădăcină este $k - 1$

a

Arbori cu rădăcină

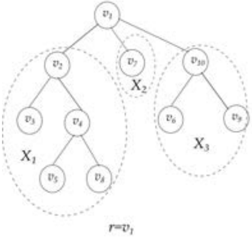
- un astfel de arbore se va numi **arbore cu rădăcină** sau **arborescență**

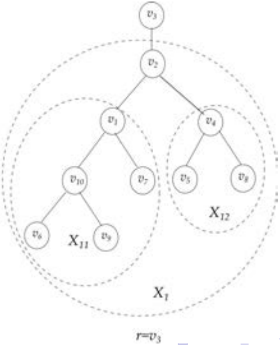


Arbori cu rădăcină

Definiție

Se numește arborescență un arbore care are un vârf r numit rădăcină, iar celelalte noduri pot fi repartizate în mulțimi disjuncte $X_1, X_2, \ldots, X_k$, $X_i \cap X_j = \emptyset$, $k > 0$, $1 \leq i < j \leq k$, $X_1 \cup X_2 \cup \ldots \cup X_k = X \setminus \{r\}$ astfel încât în fiecare din aceste mulțimi există un nod adiacent cu rădăcina, iar subgrafurile generate de acestea sunt la rândul lor arborescențe.





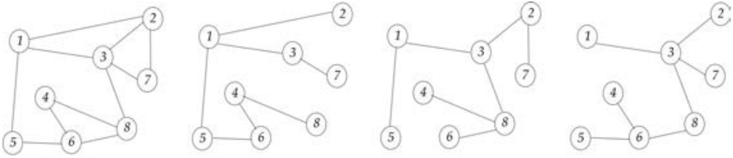
Arbori de acoperire

Fie $G = (X, U)$ un graf neorientat, conex.

Definiție

Un graf parțial $T = (X, V)$ care este arbore, se numește **arbore de acoperire** al grafului $G = (X, U)$.

- un graf conex care este arbore are un singur arbore de acoperire, el însuși
- dacă graful conex $G = (X, U)$ nu este arbore atunci admite mai mulți arbori de acoperire



$G=(X,U)$ $T_1=(X,V_1)$ $T_2=(X,V_2)$ $T_3=(X,V_3)$

Arbori de acoperire de valoare optimă

Fie $G = (X, U)$ un graf conex.

- fiecărei muchii $u \in U$ a grafului G i s-a atașat o valoare $l(u)$ care este număr real

- valoarea sau ponderea unui arbore de acoperire $T = (X, U)$ a grafului G, notată $l(T)$, este $l(T) = \sum_{u \in U} l(u)$

Problemă

Să se determine un arbore de acoperire de valoare minimă/maximă pentru un graf G conex dat.

Arbori de acoperire de valoare optimă

Algoritmul lui Kruskal (1956)

Se dă U prin lista muchiilor în ordine crescătoare a valorilor lor.

(a) Fie $V = \{u_1\}$;

(b) Pentru $k = \overline{2, m}$ execută

Dacă ($V \cup \{u_k\}$ nu conține cicluri) atunci $V = V \cup \{u_k\}$;

Sfdacă;

Sfpentru;

Observații:

- se pot parcurge muchiile grafului doar până când V conține $n - 1$ muchii; ultimele $m - n + 1$ execuții ale blocului "pentru" nu sunt necesare
- dacă muchiile nu sunt date în ordinea crescătoare a valorilor, se ordonează

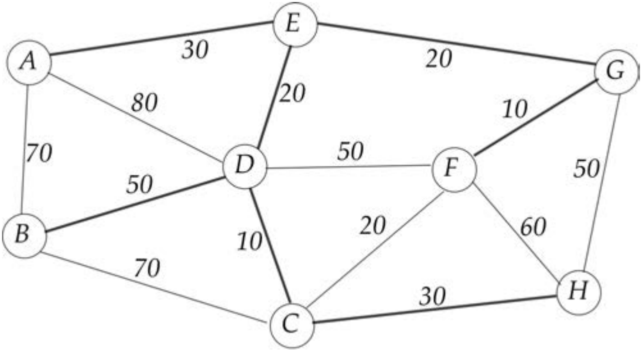
Arbori de acoperire de valoare optimă

Fie un sistem de localități $\{A, B, C, D, E, F, G, H\}$ între care există drumurile directe, cu lungimile în km, conform tabelului următor:

	A	B	C	D	E	F	G	H
A	–	70	–	80	30	–	–	–
B	70	–	70	50	–	–	–	–
C	–	70	–	10	–	20	–	30
D	80	50	10	–	20	50	–	–
E	30	–	–	20	–	–	20	–
F	–	–	20	50	–	–	10	60
G	–	–	–	–	20	10	–	50
H	–	–	30	–	–	60	50	–

- se dorește asfaltarea unor drumuri din zonă astfel încât între oricare două localități să se poată ajunge pe drum asfaltat, iar costul de asfaltare să fie minim
- problema revine la determinarea unui arbore de acoprire minimal pentru graful ce modelează problema

Arbori de acoperire de valoare optimă



a) se ordonează muchiile grafului în ordinea crescătoare a valorilor:

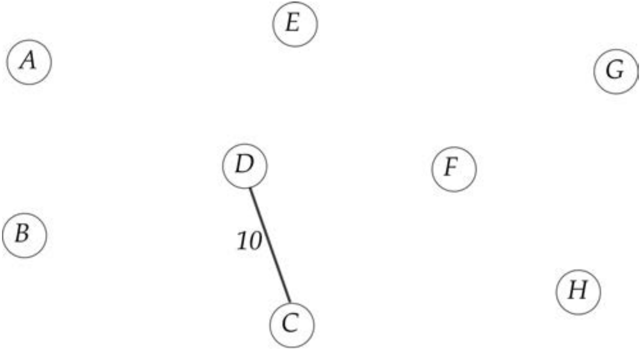
$(C, D, 10), (F, G, 10), (D, E, 20), (E, G, 20), (C, F, 20), (A, E, 30),$
$(C, H, 30), (B, D, 50), (D, F, 50), (G, H, 50), (F, H, 60),$
$(A, B, 70), (B, C, 70), (A, D, 80);$

b) se aleg muchiile, pornind de la valoarea cea mai mică astfel încât să nu se formeze cicluri: $(C, D, 10)$, $(F, G, 10)$, $(D, E, 20)$, $(E, G, 20)$, $(C, F, 20)$ nu pentru că se formează ciclul (C, D, E, G, F, E),
$(A, E, 30)$, $(C, H, 30)$, $(B, D, 50)$

și sunt selectate $m = 7 = 8 - 1 = n - 1$ muchii, deci s-a obținut un arbore de acoperire

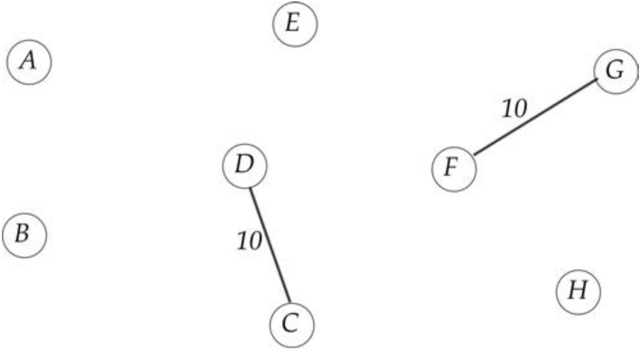
Arbori de acoperire de valoare optimă

$\underline{(C, D, 10)}$, $(F, G, 10)$, $(D, E, 20)$, $(E, G, 20)$, $(C, F, 20)$, $(A, E, 30)$, $(C, H, 30)$, $(B, D, 50)$, $(D, F, 50)$, $(G, H, 50)$, $(F, H, 60)$, $(A, B, 70)$, $(B, C, 70)$, $(A, D, 80)$;



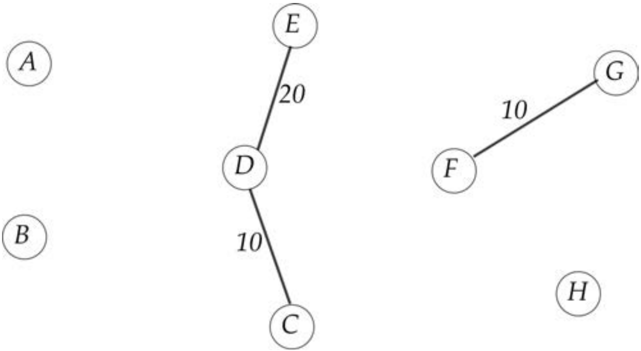
Arbori de acoperire de valoare optimă

$\underline{(C, D, 10), (F, G, 10)}, (D, E, 20), (E, G, 20), (C, F, 20), (A, E, 30),$
$\underline{(C, H, 30), (B, D, 50)}, (D, F, 50), (G, H, 50), (F, H, 60),$
$(A, B, 70), (B, C, 70), (A, D, 80);$



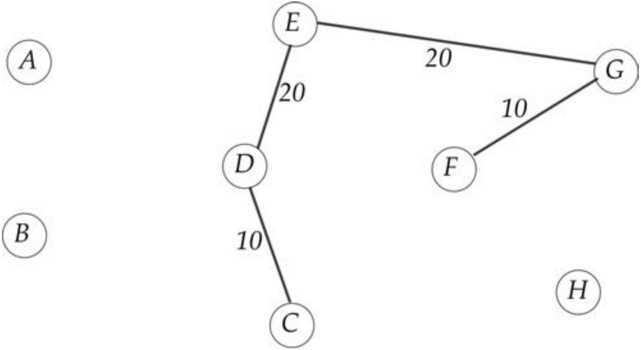
Arbori de acoperire de valoare optimă

$\underline{(C, D, 10)}, \underline{(F, G, 10)}, \underline{(D, E, 20)}, (E, G, 20), (C, F, 20), (A, E, 30),$
$(C, H, 30), (B, D, 50), (D, F, 50), (G, H, 50), (F, H, 60),$
$(A, B, 70), (B, C, 70), (A, D, 80);$



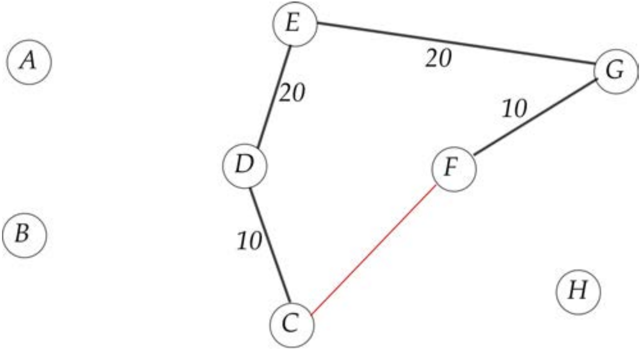
Arbori de acoperire de valoare optimă

$\underline{(C, D, 10)}, \underline{(F, G, 10)}, \underline{(D, E, 20)}, \underline{(E, G, 20)}, (C, F, 20), (A, E, 30),$
$(C, H, 30), (B, D, 50), (D, F, 50), (G, H, 50), (F, H, 60),$
$(A, B, 70), (B, C, 70), (A, D, 80);$



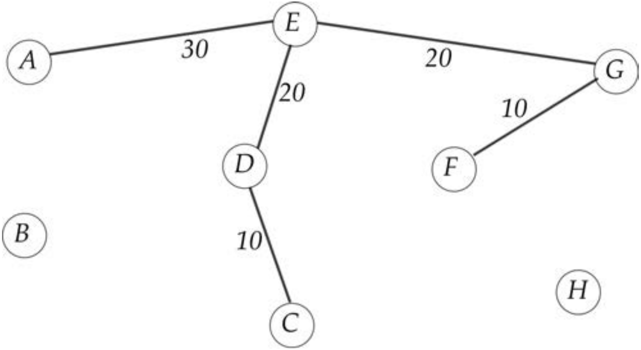
Arbori de acoperire de valoare optimă

$\underline{(C, D, 10)}, \underline{(F, G, 10)}, \underline{(D, E, 20)}, \underline{(E, G, 20)}, \underline{(C, F, 20)}, (A, E, 30),$
$(C, H, 30), (B, D, 50), (D, F, 50), (G, H, 50), (F, H, 60),$
$(A, B, 70), (B, C, 70), (A, D, 80);$



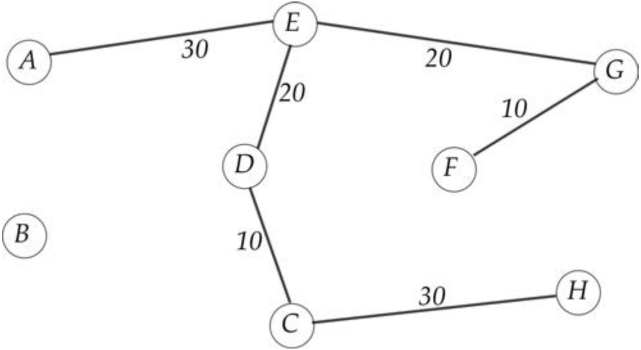
Arbori de acoperire de valoare optimă

$\underline{(C, D, 10)}, \underline{(F, G, 10)}, \underline{(D, E, 20)}, \underline{(E, G, 20)}, (C, F, 20), \underline{(A, E, 30)},$
$(C, H, 30), (B, D, 50), (D, F, 50), (G, H, 50), (F, H, 60),$
$(A, B, 70), (B, C, 70), (A, D, 80);$



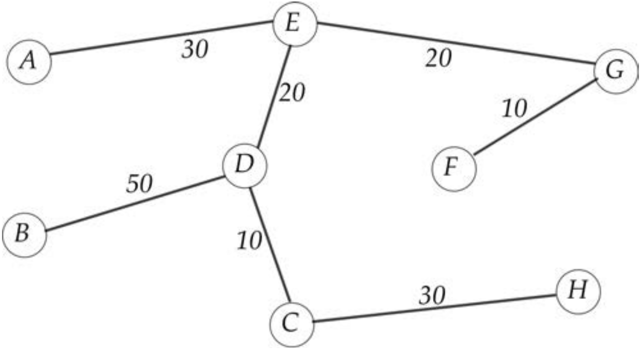
Arbori de acoperire de valoare optimă

$\underline{(C, D, 10)}, \underline{(F, G, 10)}, \underline{(D, E, 20)}, \underline{(E, G, 20)}, (C, F, 20), \underline{(A, E, 30)},$
$\underline{(C, H, 30)}, (B, D, 50), (D, F, 50), (G, H, 50), (F, H, 60),$
$(A, B, 70), (B, C, 70), (A, D, 80);$



Arbori de acoperire de valoare optimă

$\underline{(C, D, 10)}, \underline{(F, G, 10)}, \underline{(D, E, 20)}, \underline{(E, G, 20)}, (C, F, 20), \underline{(A, E, 30)},$
$\underline{(C, H, 30)}, \underline{(B, D, 50)}, (D, F, 50), (G, H, 50), (F, H, 60),$
$(A, B, 70), (B, C, 70), (A, D, 80);$



v(T)= 10+10+20+20+30+30+50= 170

- se porneste de la graful (S, V) cu S o multime cu un varf oarecare,

$V = \emptyset$,si ın $n-1$ pasi se vor adauga $n-1$ varfuri ın S ,si tot atatea muchii ın V(cate un varf si o muchie la fiecare pas)

Pseudocod:

Se dă graful $G = (X, U)$ și valorile muchiilor sale.

(a) $k = 1$; $V = \emptyset$;
    alege x din X;
    $S_k = x$;

(b) pentru $k = \overline{2, n}$ execută
    pentru $i = \overline{1, k-1}$ execută
        alege $y_i \in X \setminus S_k$ vârful adiacent lui x_i astfel încât
            muchia (x_i, y_i) are valoare minimă în $\omega(X \setminus S_k) \cap \omega(x_i)$;
    Sfpentru;
    Alege dintre muchiile selectate muchia (x_j, y_j) de valoare minimă,
        adică $l(x_j, y_j) = min\{l(x_i, y_i)|\ i = \overline{1, k-1}\}$;
    $x_k = y_j$; $S_k = S_k \cup \{x_k\}$; $V = V \cup \{(x_j, y_j)\}$;

Sfpentru.