

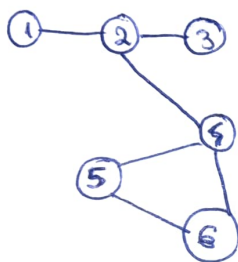
02.10.2024

Seminar 1

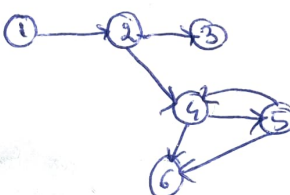
Grafuri

$$G = (V, E) \Rightarrow V = \text{vf.} \\ \Rightarrow E = \text{muchii}$$

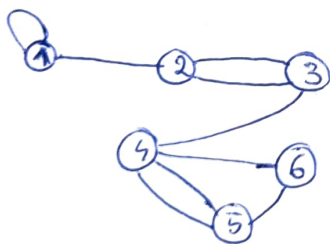
Graf neorientat:



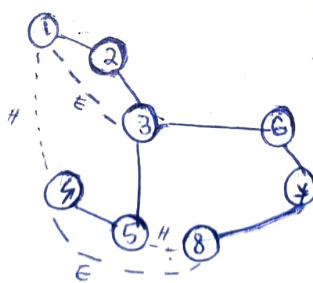
Graf orientat:



Multi graf



Exercitii:



- 7 muchii, 8 vf.
- aciclic și neorientat
- conex
- arbore (conex & aciclic)
- nu e Hamiltonian (are ciclul corect și singură dată prin toate nodurile)
- nr. muchii să fie Hamiltonian: 2
- nu e eulerian \Rightarrow nr. min eul: 2
- diametrul 5

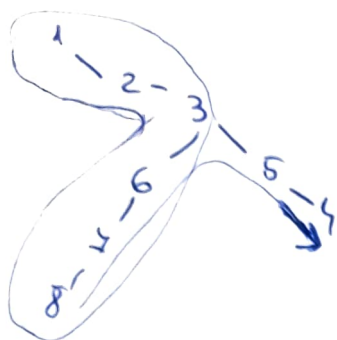
Muchie critică = dacă e eliminată, atunci graful nu mai e conex
 Dacă graful are cel puțin o muchie critică \Rightarrow nu e Hamiltonian
 Eulerian = trebuie să treacă prin fiecare muchie o singură dată și nodurile au grad par

Diametrul unui graf = cel mai lung drum

Temă: Dem. corectitudinea alg. de găsit a diametrului din arbore:

Pos 1: Parcurgere din nod care are
 Pos 2: — " — frunza cea mai departe de la pos 1

Soluția: cel mai lung drum. din al II-lea BST



CBC (camp. ~~tr~~ bicamero) : $(2, 3, 4, 5, 6)$; $(1, 2)$; $(6, 7)$; $(7, 8)$

Nod critic: 3, 5, 6, 7

Muchii critice: toate

Graf bipartit : $(1, 3, 4, 7)$ și $(2, 5, 6, 8) \Rightarrow$ bicolorare
(seamănă cu RED-BLACK TREE)

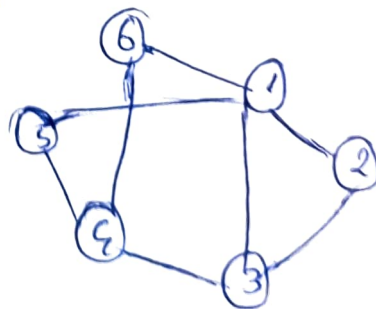
Exercițiu:

Avem m camere. Arătați că \exists 2 persoane cu același nr. de prieteni.
(nu are cum să fie $m=1$). Prieteni \Rightarrow muchie

luăm x și y 2 pers. din cei m oameni

$x - y$ sau x și y nu sunt prieteni

Ori avem med cu grad $m-1$ ori.



16.10.2024

Seminar 2Algoritm muchii critice

```

DFS(node) { // calculăm înălțimea
    vis[node] = true;
    for x in Vecini(node) {
        if vis[x] == false {
            Niv[x] = Niv[node] + 1;
            DFS(x);
        }
    }
}

```

```

int main() {
    ...
    Niv[1] = DFS(1)
    ...
}

```

// Niv[i] = înălțimea nodului i în arborele DFS

1	3	2	5	2	3	5	6	4
---	---	---	---	---	---	---	---	---

 // vector cu înălțimile nodurilor

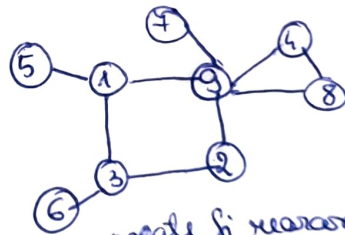
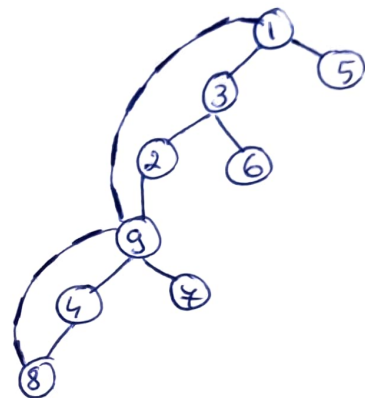
// LOW[i] = cel mai mic nivel la care poate ajunge nodul i sau un descendent al lui printr-o muchie de întoarcere

1	1	1	1	4	2	3	5	4	1
---	---	---	---	---	---	---	---	---	---

 // dacă nu are muchii înt. (1, 3 sau 9, 8) atunci se pune nivelul la care e nodul

1) Muchiile critice pot fi ~~dar~~ muchiile ^{de} arbore

2) (x, y) muchie critică $\Leftrightarrow \text{LOW}[y] > \text{NIV}[x]$



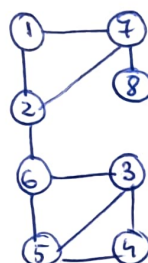
// poate fi reconstituit oricum
pt. a găsi muchiile de întoarcere Ξ

Noduri critice

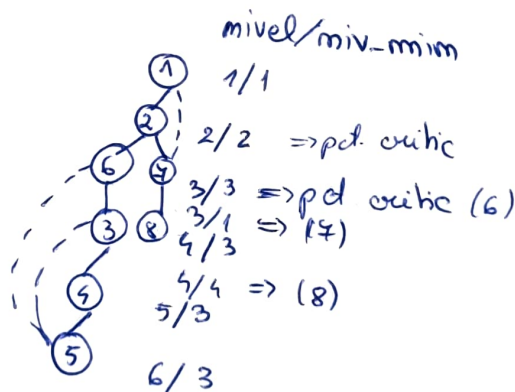
- 1) x și y noduri unde $y = \text{fin}$ în $\text{Lau}[y] \geq \text{niv}[x] \Rightarrow \text{nod critic} = x$
 (practic dacă y ~~nu~~ poate să sară peste x în vreun fel $\Rightarrow x$ nod crit.)

Componente biconexe

După fiecare punct critic găsit,
 se scot toate muchiile până întâlnești
 muchia (x, y)



S: 1 2
 2 6
 6 3
 3 4
 4 5
 5 3
 5 6



PT colociu :

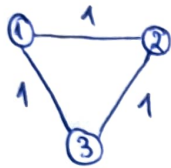
- oice materiale cod (înțeleg călărașului la ultimul col)
- norme recording, puncte pariale
- toate input testare
- putem C++ sau Python

30.11.2024

Seminar 3

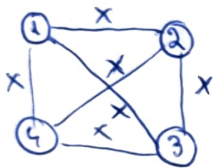
APM = arbore parțial de cost minim (negărat conex)

Câte APM-uri distincte are acest graf?



Ace 3 \Rightarrow putem elimina oricare din muchii

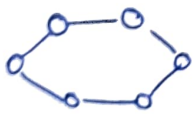
Dar acest graf?



Ace 16 \Rightarrow costul minim este $3x$ că să atingă toate nodurile

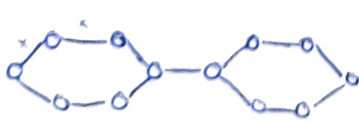
Teorema $K_5, \dots, K_m \Rightarrow$ graf complet conex cu muchiile egale (de ce și care?)

Desenați un graf cu cât mai puține noduri care are 64 de APM-uri distincte.



Nu știu dacă există altul cu 5 noduri

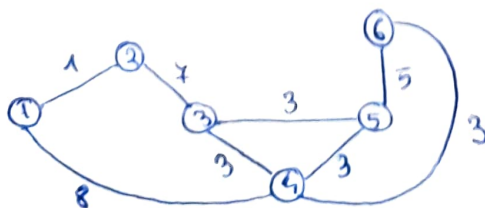
Dar pentru 42 de APM-uri?



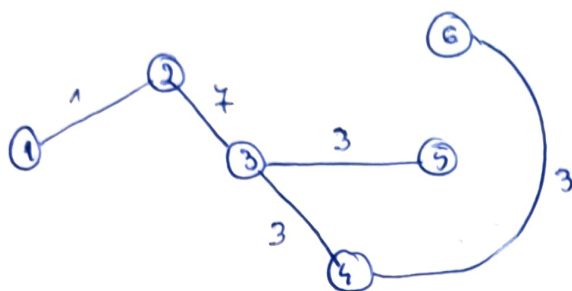
||



Obțineți un APM folosind Kruskal și unul folosind Prim.



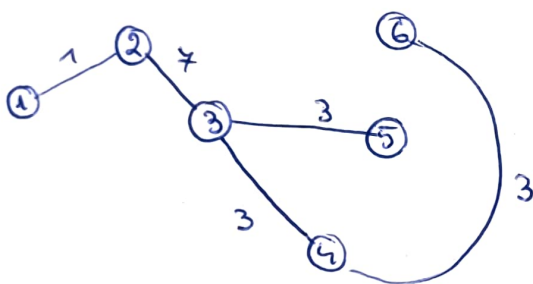
Kruskal: luăm muchiile în ordine crescătoare



$O(m \log m)$

m_1	m_2	g
1	2	1
3	4	3
3	5	3
4	5	3
4	6	3
5	6	5
2	3	7
1	4	8

Prim: luăm un mod carecare și mergem pe muchia cea mai mică. Nu includem muchia dacă formăm un ciclu în interior.



$O(m^2)$ - mod. ed.

$O(m \log m)$ - heapuri

$O(m \log m + m)$ - heapuri Fibonacci

Nu este obligatoriu ca cele 2 alg. să dea același APM

Se dă un graf, cum det. cât mai eficient toate muchiile care se află într-un ciclu?

Ne folosim de DFS \Rightarrow au muchiile de întoarcere.

Orice muchie critică e în APM.

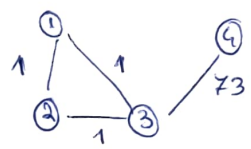
20.11.2024

Seminar 4

① Fie G = graf conex cu e muchie de cost minim.

a) Arătați că există un APM care îl conține pe e

de exemplu:



să zicem $e = (2, 3)$

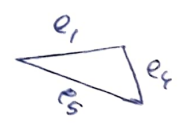
Facem APM-ul, luăm e ca să închidă ciclul și eliminăm oricare altă muchie din ciclu

b) a vorbit despre el

② Dem. că dacă un graf are valori distincte are un unic APM

$e_1^{(3)}, e_2^{(5)}, e_3^{(11)}, e_4^{(21)} \dots$
 $e_1^{(3)}, e_2^{(15)}, e_3^{(14)}, e_4^{(18)} \dots$

$\Rightarrow e_4, e_5 + \alpha \text{ etc.} \Rightarrow \text{ciclu (?)}$



(vine din Kruskal)

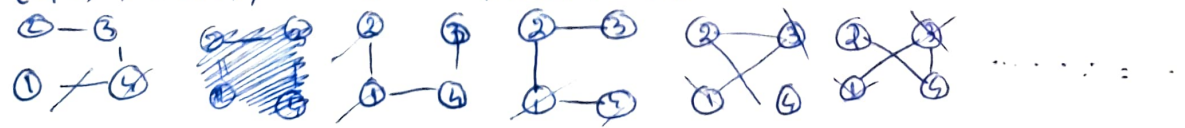
③ Din tema de data trecută : K_m

Ne folosim de Prüfer (secvență)

$P = (p_1, p_2, \dots, p_{m-2})$ secvența formă cu nivelul cel mai înalt.

Pt $m=4$. Pornim din 1 și îl eliminăm

$(2, 3); (4, 3); (1, 4); (1, 2); (3, 2); (3, 4)$.



④ Se dă un graf G și T un APH, unde $G = (V, E, w)$. Avem un graf $G' = (V, E, w')$ cu $w'(e) = w(e) + k$. Este T APH și pentru G' ? Da.

T - AP în G'

notăm $\sum_G(T') = k(m-1) + \underbrace{\sum_G(T)}_{\geq \sum_G(T)}$ suma costurilor

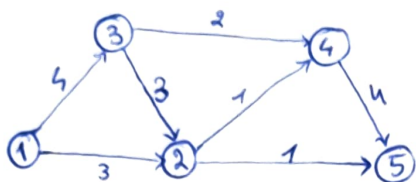
și $\sum_{G'}(T') \geq \sum_{G'}(T)$

04.12.2024

Seminar 5

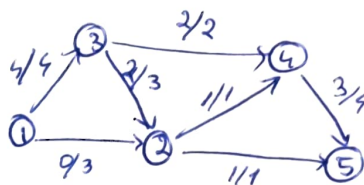
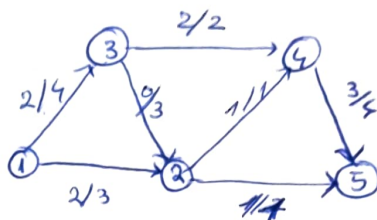
-flux-

①

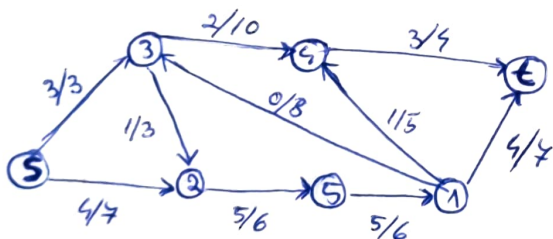


max flow = min cut

aici taietura ar fi prin muchiile: $(3,4); (2,4); (2,5)$

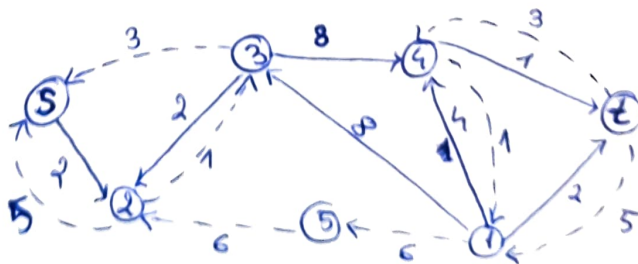


②

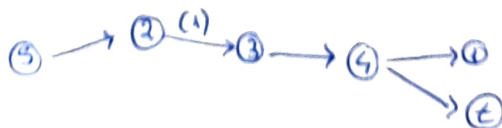


Drumul de crestere: $s \rightarrow 2 \xrightarrow{(1)} 5 \rightarrow 1 \rightarrow t$

Graful residual:

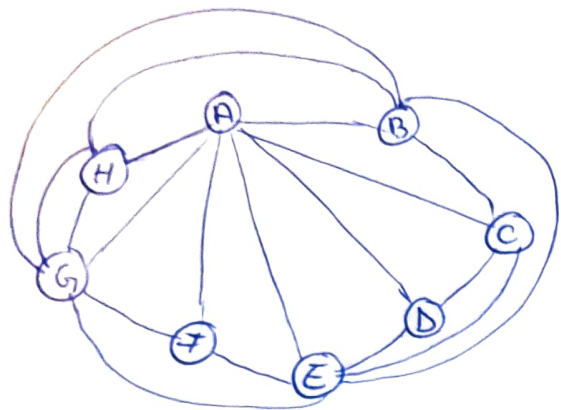


Facem BFS:



④

Graf planar = dacă pot să-l desenez într-un mod în care nu se „o”



=> este planar

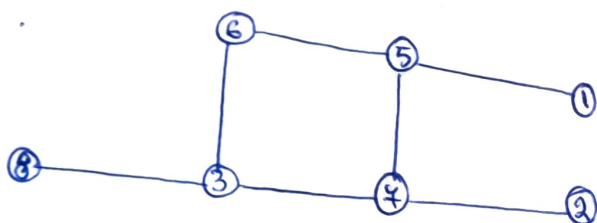


pot să-l desenez

TEMA: desenați un graf cu 10 noduri, 15 muchii și aplicați algoritmul de 6 colorare pe el.

Seminar 6

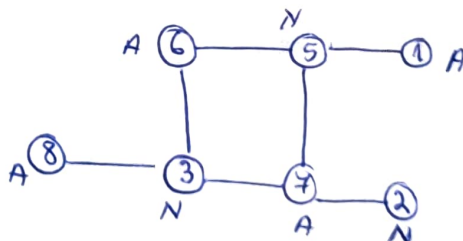
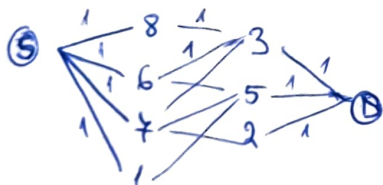
- ① Care e cuplajul maxim? Cum arată de asociată grafului?



$\left. \begin{matrix} 1-5 \\ 2-4 \\ 3-8 \end{matrix} \right\}$ muchiile alise pentru cuplaj

cuplaj = micim mod nu are mai mult de 2 muchii conectate

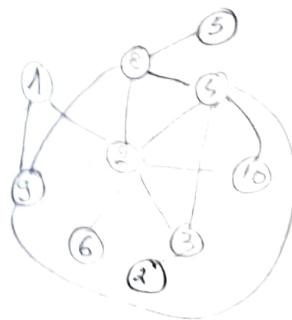
②



- ③ Este un sir grafic? (dacă astea pot fi gradele unor noduri)

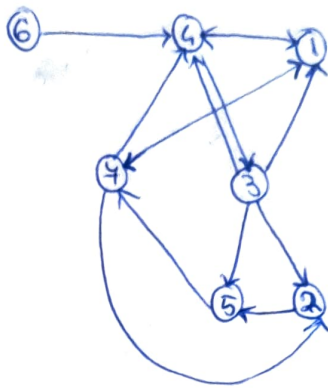
1 2 3 4 5 6 7 8 9 10
 2 4 3 5 1 2 1 4 4 3

Folosim alg. Hall-Hakimi \Rightarrow luăm model
 cu cel mai mare grad și îi punem
 vecinii ș.a.m.d. (dar scădem din vecini
 și mereu alegem din graf, nu din sir)

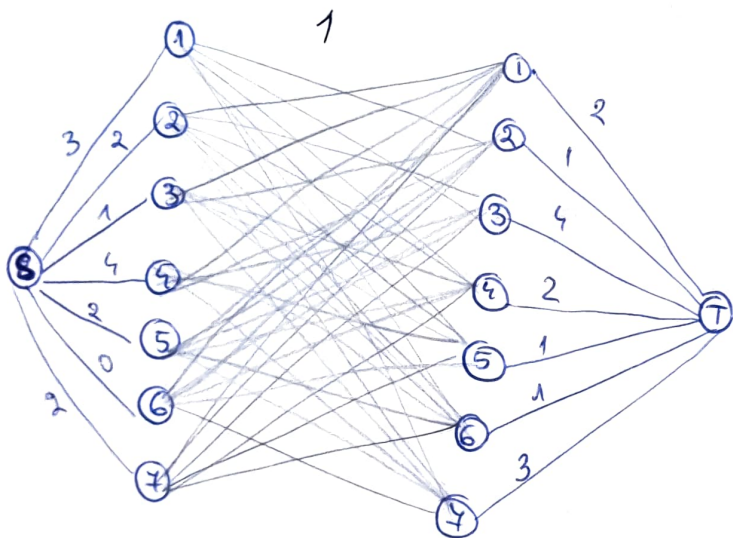


④ Un grup de prieteni care s'au oșit:

out in
 1: 3 2
 2: 2 1
 3: 1 4
 4: 4 2
 5: 2 1
 6: 0 1
 7: 2 3
 mod / nr prieteni



⑤ Cum transformăm ex 4 într-o rețea de transporturi? (1) facem flux
 Dublăm nodurile! Adăugăm S și T! Toate muchiile între nodurile
 dublate au costul 1!



⑥

	1	2	3	4
A	0			6
B				
C		c	b	
D			e	7
			g	

Tablă de șah. Nr. max. de ture care să
 rămână pe tablă de șah? 4

