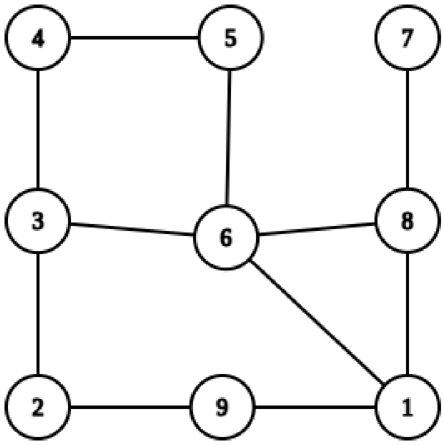


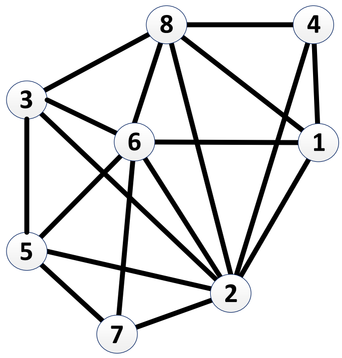
Pentru graful din imaginea din stânga:

1) Care sunt nodurile critice ?

2) Care sunt muchiile critice ?

3) Exemplificati cum funcționează bf(9) până când sunt vizitate 6 noduri, ilustrand si arborele bf asociat; vecinii unui vârf se consideră în ordine lexicografică

4) Puneți ponderi pe muchii astfel încât costul unui arbore parțial de cost minim în graful obținut să fie 66.

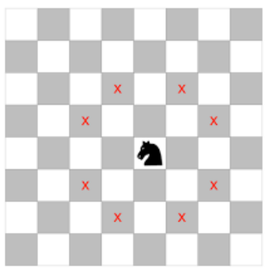


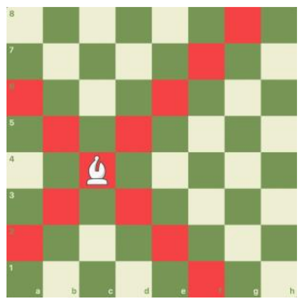
5) Care este distanta de editare intre cuvintele “examen” si “marire” ? Justificați

6) Descrieți algoritmul de 6-colorare a vârfurilor unui graf neorientat conex planar și **exemplificați** acest algoritm pentru graful alăturat. Justificați și de ce acest graf este planar.

Barem 0.5 fiecare problema 1)-6)

7) După un examen obositor, Schiorel a decis să se relaxeze jucând o partidă de șah . După ce a pierdut vreo 10 jocuri la rând a zis că era timpul să învețe și **un pic de teorie** (puteți încerca și voi asta), și a început să studieze mutarile calului (în L) și ale nebunului (în diagonală). Tabla are marimea **p*q**.





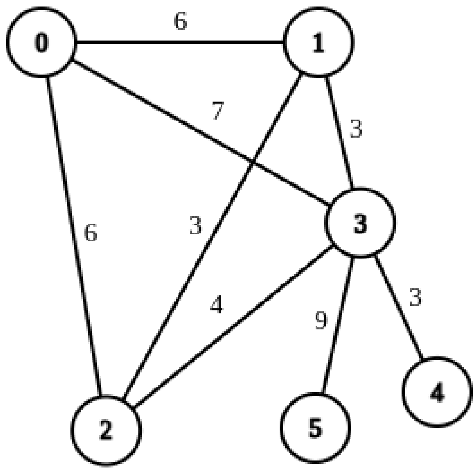
După o perioadă a decis să inventeze o piesă calo-nebunul, calo-nebunul putea muta și ca un cal și ca un nebun, totuși costul pentru a muta ca un cal era C și cel de a muta ca un nebun era B.

Schiorel încearcă să-și dea seama cum poate să mute Calo-Nebunul de la poziția S la poziția F cu cost (total) minim. Pe tabla există și obstacole (pătrățele pe care piesa nu se poate opri).

Descrieți cum puteți rezolva aceasta problemă și complexitatea soluției. Dacă exista mai multe soluții/implementării puneți accent pe discuția despre când ar trebui sa folosim o soluție si când alta.

Barem: **1,5p** (0,75 soluție corectă + 0,75 discuții complexitate + complexitate optimă)

Pentru graful din stânga (vecinii se consideră în ordine lexicografică):



1) Exemplificați Dijkstra din 4, opriți-va după ce ați găsit distanța către 0

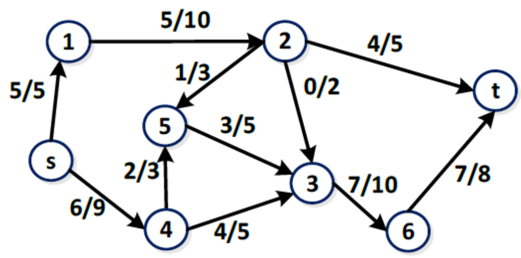
2) Cum funcționează algoritmul lui Prim din 2 ? Exemplificați alegerea primelor 6 muchii.

3) Este graful bipartit ? Dacă nu eliminați un număr minim de muchii astfel încât el sa devina bipartit. Care este numărul maxim de muchii ale unui graf bipartit cu 8 vârfuri? Justificați.

4) Există lanț eulerian în graf? Dacă nu adăugați număr minim de muchii astfel încât graful format sa admită lanț eulerian, descriind și strategia după care ați adăugat muchiile. Indicați un lanț eulerian în graful obținut. Enunțați o condiție necesară și suficientă ca un graf neorientat să aibă un lanț eulerian.

0.5p fiecare problema 1)-4)

5) Definiți noțiunile de flux, tăietură, tăietură minimă și lanț nesaturat/drum de creștere. Ilustrați pașii algoritmului Ford-Fulkerson pentru rețeaua din figura următoare (pe un arcul e sunt trecute valorile f(e)/c(e) reprezentând flux/capacitate), pornind de la fluxul indicat și alegând la fiecare pas un s-t lanț f-nesaturat de lungime minimă (algoritmul Edmonds-Karp). Indicați o tăietură (s-t tăietură) minimă în rețea (se vor indica vârfurile din bipartiție, arcele directe, arcele inverse) și determinați capacitatea acestei tăieturi. Mai există și o altă s-t tăietură minimă în această rețea? Justificați răspunsurile. (**1p)**



6) a) Dați exemplu de un graf planar conex care are o hartă având cel puțin două fețe de grad 4 și o hartă care nu are fețe de grad 4.
b) Fie M=(V, E, F) o hartă conexă cu n>6 vârfuri și m muchii cu gradul minim al unui vârf egal cu 4. Arătați că m ≤ 3n - 6 și M are cel puțin 6 vârfuri de gradul mai mic sau egal cu 5.
(1.5p)

Lăzăroiu Teodora - Bianca, 241

Examen Algoritmi Fundamentali

Partea I

5. examen → marire

mxamen

mamen

mar en

mari en

marinen

marire

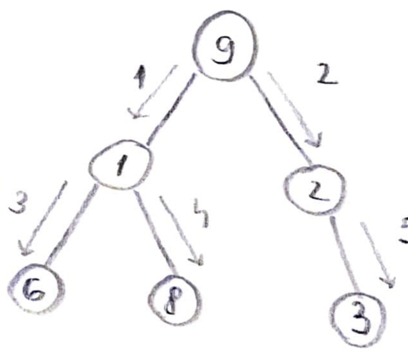
=> distanța de editare e 6

1. noduri critice: 8, 3, 5, 1

2. muchii critice: 7-8

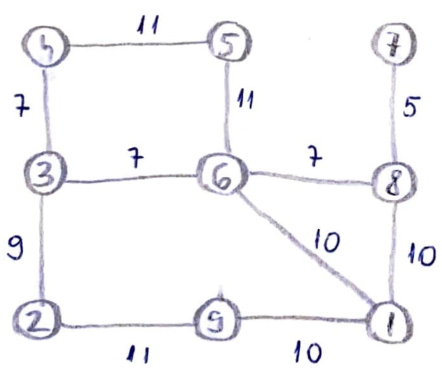
3. bf(9) pleacă din vârful 9 și vizitează vecinii săi 1 și 2

Din 1 vizitează 6 și 8 apoi din 2 vizitează nodul 3

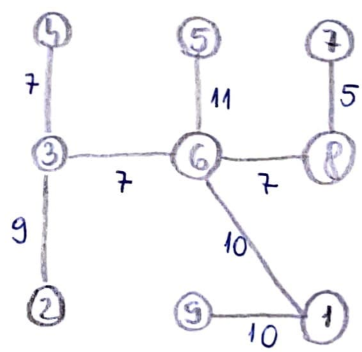


bf(9): 9, 1, 2, 6, 8, 3

4.



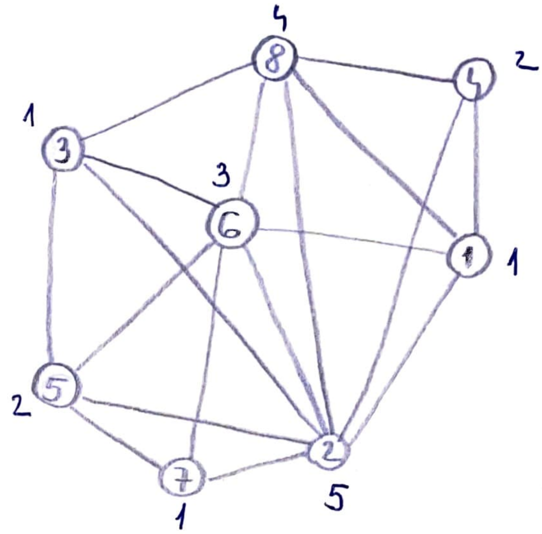
apcm →



6. Orice graf planar și conex este 6 - colorabil

Algoritm: dacă numărul de vârfuri este ≤ 6 atunci se colorează câte un vârf cu o culoare în caz contrar se alege un vârf cu gradul ≤ 5 și se colorează cu una dintre cele 6 culori pe care nu o are deja un vecin

Colorăm vârfurile în ordine crescătoare dacă gradul vârfului ≤ 5. altfel nodul respectiv va fi colorat la final



ordinea de colorare a nodurilor: 1,3,4,5,6,7,8,2

{1,2,3,4,5,6} - culorile

Gradul unui nod scade în momentul în care am colorat un nod vecin cu acesta

Graful este planar întrucât poate fi rearanjat ca muchiile sale să nu se intersecteze decât acolo unde există un nod

7. În cazul de față se poate folosi algoritmul lui Prim pentru construirea arborelui parțial de cost minim. Algoritmul lui Prim construiește drumul de cost minim din aproape în aproape, spre deosebire de alg. lui Kruskal, ceea ce este realist și corespunde cu mișcările unei piese de șah

La fiecare pas facem o mișcare de cost minim și verificăm dacă aceasta este posibilă (dacă nu avem un obstacol sau spațiul se află pe tablă)

Lăzăroiu Teodora - Bianca

Examen Algoritmi Fundamentali

Partea II

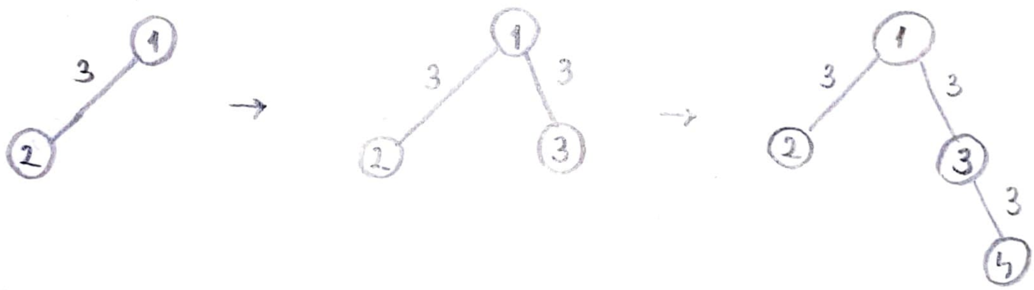
1. drumul minim de la 4 la 0 — -1 = nu are tată

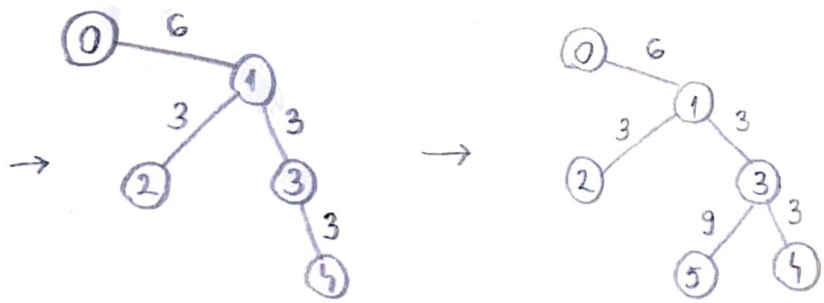
d/tata	0	1	2	3	4	5
	∞/-1	∞/-1	∞/-1	∞/-1	0/-1	∞/-1
4 :	∞/-1	∞/-1	∞/-1	3/4	-	∞/-1
3 :	10/3	6/3	7/3	-	-	12/3

↳ distanța minimă : 10

4 - 3 - 0

2. algoritmul lui Prim din 2



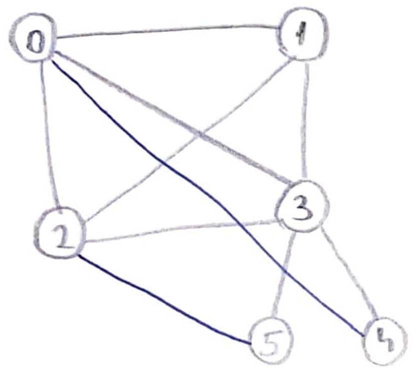


cost minim : 24

un arbore parțial are n-1 muchii => 6-1=5 muchii

4. condiție necesară și suficientă ca un graf neorientat să aibă un lanț eulerian: să aibă cel mult 2 vârfuri de grad impar

avem toate vârfurile de grad impar deci nu avem lanț eulerian ca să adăugăm număr minim de muchii vom lăsa 2 noduri cu grad impar deci trebuie să creștem gradul cu o unitate la 4 dintre noduri => unim 2 noduri și alte 2 noduri din graf



am unit 2-5 și 0-4

lanț eulerian: 3-2-5-3-0-1-2-0-4-3-1

5. Flux: un flux într-o rețea de trasport $N=(G,S,T,i,C)$ este o funcție $f: E \to \mathbb{N}$ cu proprietățile:

1) $0 \le f(e) \le c(e), \forall e \in E(G)$ condiția de mărginire

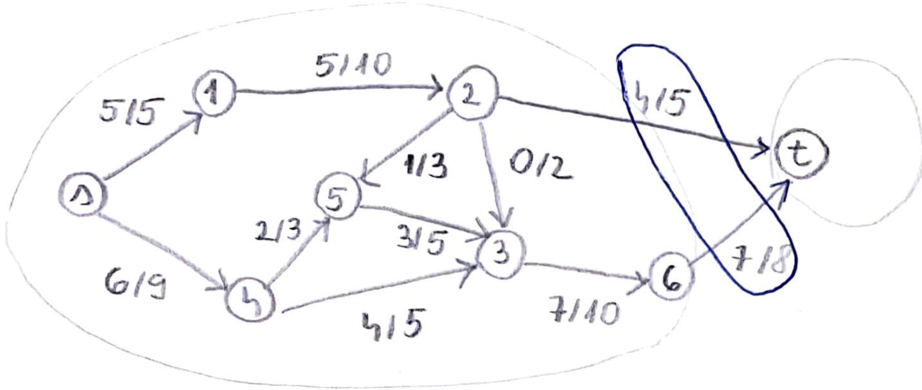
2) Pentru orice vârf intermediar v

$$\sum_{uv \in E} f(uv) = \sum_{vu \in E} f(vu)$$ condiția de conservare a fluxului

Tăietură: Fie $N=(G,S,T,i,C)$ o rețea. O s-t tăietură $K=(X,Y)$ în rețea este o bipartiție (X,Y) a mulțimilor vârfurilor V astfel încât $s \in X$ și $t \in Y$

Tăietură minimă: Fie N o rețea. O tăietură $\tilde{K}$ se numește tăietură minimă în N dacă $c(\tilde{K}) = \min\{c(K) \mid K$ este tăietură în $N\}$

Lanț nesaturat: un s-t lanț P se numește lanț nesaturat sau drum de creștere dacă $i(P) \neq 0$ unde $i(P)$ = capacitatea reziduală a lanțului



K = tăietură minimă

$C(K) = 5 + 8 = 13$

bipartiție: - s, 1, 2, 3, 4, 5, 6 (partea 1)
- t (partea 2)

nu mai există o altă tăietură minimă:

$9 + 5 = 14 > 13$

$9 + 10 = 19 > 13$

$10 + 3 + 5 = 18 > 13$

$10 + 3 + 5 + 5 = 23 > 13$

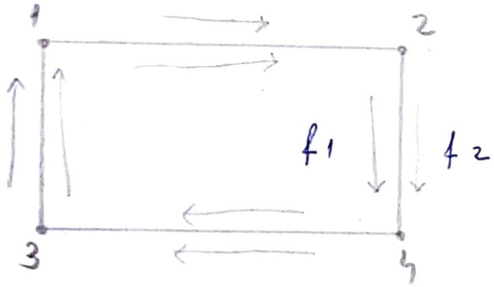
$10 + 10 = 20 > 13$

$5 + 5 + 5 = 15 > 13$

$5 + 10 = 15 > 13$

$5 + 8 = 13$ minim

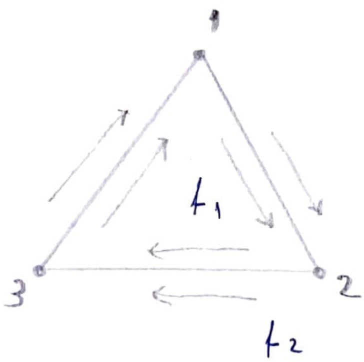
6.a.



$d(f_1) = 4$

$d(f_2) = 4$

are 2 fețe de grad 4



$d(f_1) = 3$

$d(f_2) = 3$

nu are fețe de grad 4

b. Proprietate graf planar:

$G = (V, E)$ conex cu $n = |V| > 2$ și $m = |E|$

atunci $m \leq 3n - 6$, $\exists\, x \in V$ cu $d(x) \leq 5$

5. fluxul mai permite:

$s \xrightarrow{7/9} 4 \xrightarrow{5/5} 3 \xrightarrow{8/10} 6 \xrightarrow{8/8} t$

$s \xrightarrow{8/13} 4 \xrightarrow{3/3} 5 \xrightarrow{4/5} 3 \xrightarrow{1/2} 2 \xrightarrow{5/5} t$

redirecționăm fluxul

flux maxim: 13