FPTAS (Fully polynomial time approximation scheme)

Timp polin. în $\frac{1}{\varepsilon}$, aprox. $1+\varepsilon$

ALG – K · profit(S)

profit (o) – K · profit (o') $\le$ m · K
profit (S) $\ge$ K · profit (o')

$\Rightarrow$ K · profit (S) $\ge$ profit (o) – m · K
$\ge$ profit (o) – ~~m~~ · $\frac{\varepsilon P}{\cancel{m}}$ $\ge$

$\ge$ profit (o) – εP
profit (o) $\ge$ εP

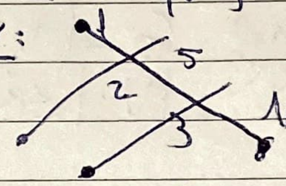
$\Rightarrow$ K · profit (S) $\ge$ profit (o) – ε · profit (o)
$\ge$ $(1-\varepsilon)$ profit (o) OPT

K – center

Se dă un graf $G=(V,E)$ și o metrică de pe graf și un nr. K. Să se găs. o mț. din K puncte care minimiz. distanța de la un punct la cele mai apropiate puncte din cele K. Căutăm o mț. $V' \subseteq V$ cu $|V'| \le K$.

$$\min \max_{v \in V} \min_{u \in V'} d(u,v)$$

ex:



$K=2$, OPT$=3$

Alg. 2-aprox pt. K – center

for $i = 1$ to m → calc. G_i^2 (unde G_i este graful cu primele i muchii)
→ calc. un maximal independence set I în G_i^2
→ dc. $|I| \le K$ at. STOP return cost (e_i)

Fie i^* pasul la care se oprește alg.; OPT $\ge$ cost (e_{i^*})

$|I| < 6$

min — cost (e_{i^*}) — OPT — 2 cost (e_i)

Dc. avem un ~~d~~ independence set de mărime $\le K$ pe G_i^2 $\Rightarrow$ dom (G_i) $\le$ K, dar pt $\forall i \le i^*$ $\Rightarrow$ este independence set pe G_i^2 $\ge$ K

Mț. I din G_i^2 este și un dominating set în G_i^2.
Dar în G putem conecta toate nodurile din graf la I printr-o distanță.

Seminarul 5
Prog. alg. ef.

1) Se dă un graf $G = (V, E)$ și un nr. k.
Vrem să găsim o mulț. $V' \subseteq V$ de cardin. k a.î. să fie minimizată ~~max~~ $\max_{v \in V} \min_{u \in V'} d(u, v)$

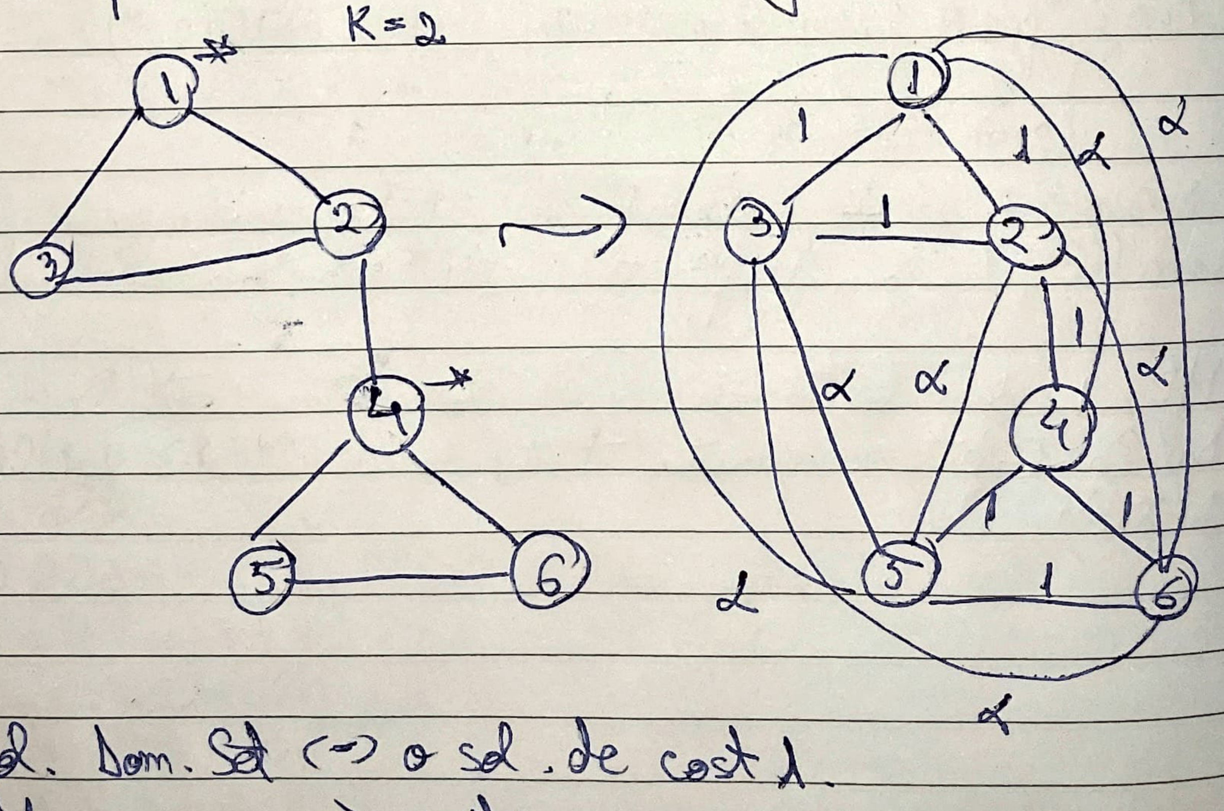
Dc. d este o metrică at. avem o 2-aprox. Ar. că dc. d nu e metrică nu putem avea o α-aprox. pt. orice α.

→ Dominating Set

Se dă un graf $G = (V, E)$ și nr. k
$\exists$ $V' \subseteq V$, $|V'| \leq k$ a.î. $\forall$ $v \in V$ avem un $x \in V'$ cu ~~v~~ $(x, v) \in E$?

Vrem să red. pb. de mai ↑ cu Dominating Set.

$k = 2$

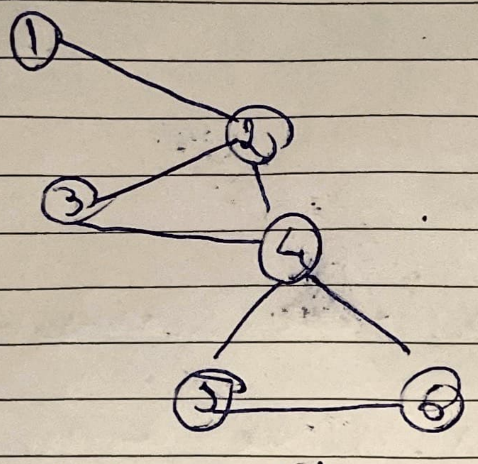


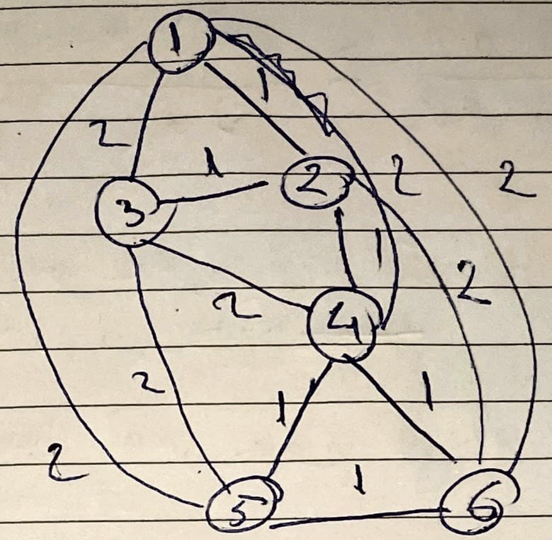
⇒ O sol. Dom. Set ⟺ o sol. de cost 1.

DA ⟶ cost 1

NU ⟶ cost α

2) Ar. că dc. d este metrică, nu putem avea 2 - ε aprox.





DA → 1
NU → 2

3) Pb. rucsacului
~~Ar. că alg. nu e bun pt. aprox. Greedy~~
Pt. pb. rucsacului avem alg. care sortează obiectele în funcție de $\frac{\text{profit}}{\text{size}}$ și pune obiectele în rucsac în ac. ordine.
Ar. că raportul dintre sol. returnată de ac. alg. și optim poate fi arbitrar mai mare

profitul maxim

$B(\text{dim. rucsacului}) = 2m - 1$
$\text{size}(a_1) = 2m-1$
$\text{profit}(a_1) = 2m$
$\text{size}(a_2) = 1$
$\text{profit}(a_2) = 2$

$a_2 = \frac{2}{1}$

$a_1 = \frac{2m}{2m-1}$

$a_1, a_2, \dots, a_K$

$$\frac{profit(a_i)}{size(a_i)} \geq \frac{profit(a_{i+1})}{size(a_i)}$$

Fie K cel mai mare nr. a.î.

$\cancel{size}$ $size(a_1) + size(a_2) + \cancel{size}(\dots + size(a_K) \leq B$

Alg. alege cea mai bună var. dintre $\{a_1, a_2, \dots, a_K\}$, $\{a_{K+1}\}$.

→ Ar. că alg. e 2-aprox.

$$OPT \leq \underbrace{profit(a_1) + \dots +}_{până\ de\ K} \underbrace{profit(a_{K+1})}_{a_K}$$

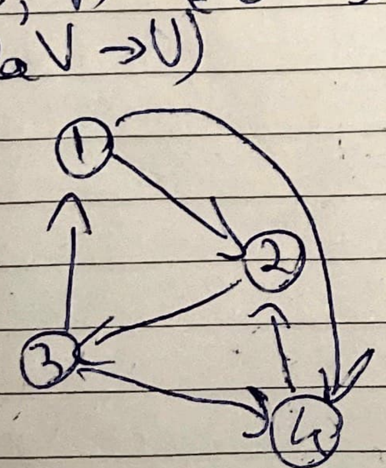
⇒ nu mai pot fi puse în rucsac

⇒ cel mai bun e cel care are $\frac{OPT}{2}$

4) Feedback Vertex Set pe graful turneu

Se dă un graf orientat $G = (V, E)$ în care pt. orice $U, V \in V$ avem $(U, V) \in E$ sau $(V, U) \in E$ (orice de la $U \to V$ ori arc de la $V \to U$)

Ex:

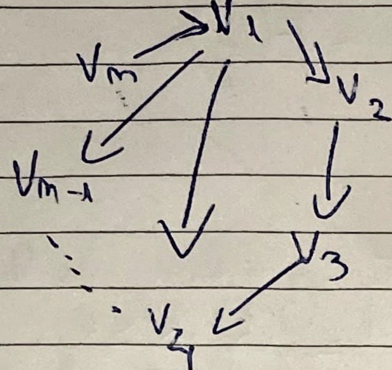


Să se găs. o mţ. de cardinalit. minimă a.î. graful obţ. prin elim. lui V' să fie aciclic.

+ că e 3-aprox.

G are un ciclu (=) are un ciclu de lung. 3

f-aprox pt. Set Cover când $f = 3$



$U = \{$ ciclurile de lungime 3 $\}$

$S_i = \{$ ciclurile care conțin nodul $V_i \}$

cicluri de lung. 3

Cursul 10

Alg: alg. ef.

Algorithm fixed parameter

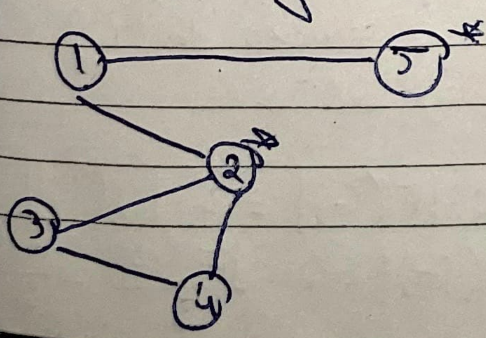
Def: O pb. (I, k) s.n. „fixed parameter tractable" dc. admite un alg. de complexitate $f(k) \cdot poly(n)$, unde k este un parametru al pb. independent de n (= mărimea inputului).

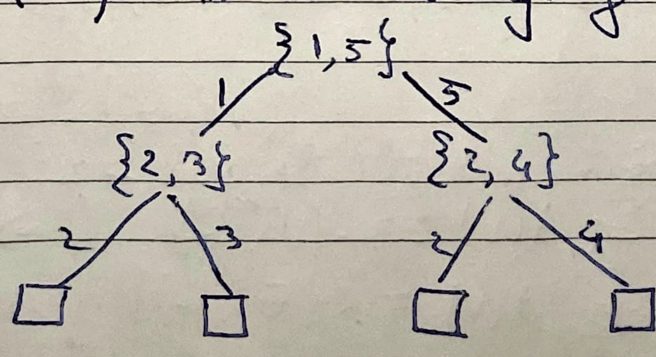
$$O^*(f(k)) = O(f(k) \cdot poly(n))$$

Un alg. fixed parameter de complexitate $O^*(2^k)$ pt. pb. Vertex Cover.

Def: Vertex Cover - Se dă un graf neorientat $G = (V, E)$ și un nr. k. Să se det. dc. $\exists V' \subseteq V$, $|V'| \leq k$ a.î. $\forall (a, b) \in E$, $a \in V'$ sau $b \in V'$.

$k = 2$ Brute force. $O^*(n^k)$ - nu este un alg. fixed parameter





Vertex Cover (G, k) $O(2^k \cdot m)$

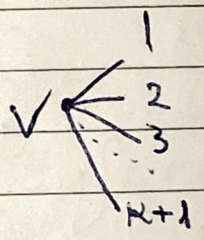
{ Dacă $G \neq \emptyset$ şi $k = 0$ => ~~return~~
Dc. $G = \emptyset$ => am găsit un VC
Alegem arbitrar o muchie $\{a, b\} \in E(G)$
VertexCover($G \setminus \{a\}, k-1$)
VertexCover($G \setminus \{b\}, k-1$)
}

Un kernel este o transformare în timp polin. de la o instanță (i, k) la o instanță (i', k'), unde $|i'| \leq f(k)$, $k' \leq k$.
(i, k) este adev $\Leftrightarrow$ (i', k') este adevărată

$\boxed{\text{Algoritm FPT} \Rightarrow \text{Kernel}}$

Alg. FPT = avem un alg. de complexitate $f(k) \cdot n^c$ pt. o constantă c.
Avem 2 cazuri: 1) $n \geq f(k)$ => alg. e polin. pt. că are complex. $O(n^{c+1})$
=> putem decide dc. (i, k) este YES/NO
2) $n \leq f(k)$ => este kernel

$\boxed{\text{Kernel} \Rightarrow \text{FPT}}$ Brute force



Kernel pt. Vertex Cover

I Dc. $\exists v \in V$ cu $d(v) = 0$ => $(G, k) \rightsquigarrow (G \setminus \{v\}, k)$
II Dc. $\exists v \in V$ cu $d(v) > k$ => $(G, k) \rightsquigarrow (G \setminus \{v\}, k-1)$
$|V(G)| \leq k(k+1)$
III Dc. $\exists v \in V$ cu $d(v) = 1$ => $(G, k) \rightsquigarrow (G \setminus \{u\}, k-1)$ unde $(u, v) \in E$
(îl putem scoate și pe v)

După primele 3 reguli, avem $|V(G)| \leq |E(G)|$ – nr. de vf. din graf ≤ nr. de muchii din graf

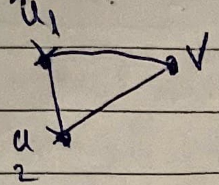
$$|V(G)| \leq |E(G)| \leq k^2$$

Scop: să elimin nodurile de grad ≥ 2

Dc. reușesc asta: $\sum_{v \in V} d(v) \geq 3 \cdot |V(G)|$, $\sum_{v \in V} d(v) = 2 \cdot |E(G)|$ } => $2 \cdot |E(G)| \geq 3 \cdot |V(G)|$

$$|V(G)| \leq 2 \cdot \frac{|E(G)|}{3} \leq \frac{2k^2}{3}$$

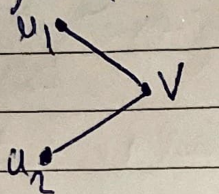
IV a) $d(v)=2$



$\rightsquigarrow$ vecinii au și ei muchie între ei

$\rightsquigarrow (G \setminus \{u_1, u_2, v\}, K-2) \rightarrow$ trei noduri mai puțin și 2 în VC

IVb) $d(v)=2$



$\rightsquigarrow$

$u = u_2 = u_2 \rightsquigarrow (G \setminus \{u_1, u_2, v\} \cup \{u\}, K-1)$

Vreau să arăt că G are o soluție cu K noduri $\Leftrightarrow (G \setminus \{u_1, u_2, v\} \cup \{u\})$ are o sol. cu $K-1$ noduri.

$\Leftarrow$ u este în soluție $\Rightarrow$ pun u_1, u_2 în soluție

$\Leftarrow$ u ~~este~~ nu este în soluție $\Rightarrow$ pun v în soluție

$\Rightarrow$ u_1 sau u_2 sunt în sol. $\Rightarrow$ pun u în sol.

$\Rightarrow$ nici u_1, nici u_2 nu sunt în sol. $\Rightarrow$ nu pun u în sol.

Cursul 11

Prog. alg. ef.

Algorithm fixed parameter

1. "Crown" decomposition

Def: O „crown decomposition" a unui graf neorientat $G=(V,E)$ este o partiționare a nodurilor $V = I \cup H \cup C$ a.î.:

a) nodurile din I sunt independente

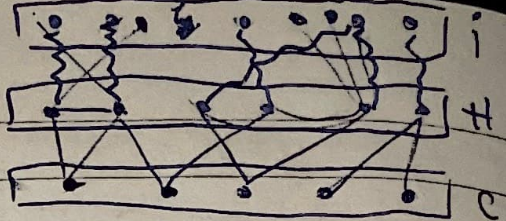
b) $\exists$ un cuplaj între nodurile din H și cele din I care acoperă toate nodurile din H

c) $\nexists$ muchie între noduri din I și noduri din C

Lemă: Dându-se un graf neorientat $G=(V,E)$ și un nr. K avem unul din urm. 3 cazuri (pe care le putem det. în timp polin.):

1) G are un cuplaj de mărime $K+1$ ($\rightsquigarrow$ NU pt. desen)

2) G are o descompunere „crown" $\rightsquigarrow (G,K)$ reducem $(G \setminus I \cup H, K-|H|)$



3) are cel mult 3K noduri

<u>Coldar</u>: Putem găsi un kernel de mărime 3K pt. Vertex Cover

n. de muchii din cuplaj → X este mult. nodurilor din M

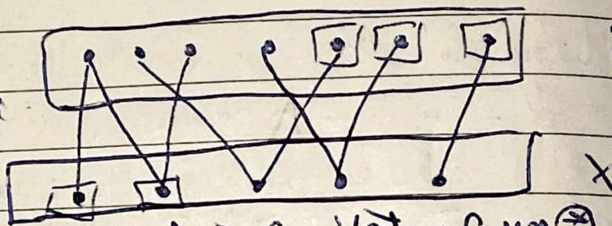
<u>Dem.</u>

n. de muchii din cuplaj → X este mult. nod. din M

Fie M un cuplaj maximal în G. Dc. $|M| \geq K+1$ ⇒ cazul 1 (→ cut)

i este o mult. indep.

Considerăm graful bipartit det. de i și X.



noduri în Vertex Cover ⊛

König, 1931: Într-un graf bipartit, minimum Vertex Cover = maximum matching.

Rezolv exact Vertex Cover pe graful bipartit det. de i și X.
Fie W Vertex Coverul optim. Avem 2 cazuri:
1) $W \cap X \neq \emptyset$ ⊛ ⇒ Crown Decomposition $H = W \cap X$

i = capetele

C = ~~tot~~ restul nodurilor

2) $W \cap X = \emptyset$ ⇒ $W = i$ ⇒ cel mult 3K noduri (caz 3)

<u>Iterative Compression</u>

Se dă (G, k). Are G un Vertex Cover (VC) de mărime $\leq k$?

1. Rulăm alg. de 2-aprox. pt. VC. Dc. ~~A~~LG $\geq 2K+1$ ⇒ NU

$ALG = |Z|$, unde Z este VC returnat de algoritm

Notăm cu X' soluția optimă.

2. Dc. știm $Z \cap X$, putem det. restul

$N(Z)$ = nodurile din vecinătatea lui Z adică

$N(Z) = \{a \in V \mid (a,x) \in E,\ x \in Z\}$

$Z_1 = Z \cap X$, at. $X = N(Z \setminus (Z \cap X)) \cup (Z \cap X)$

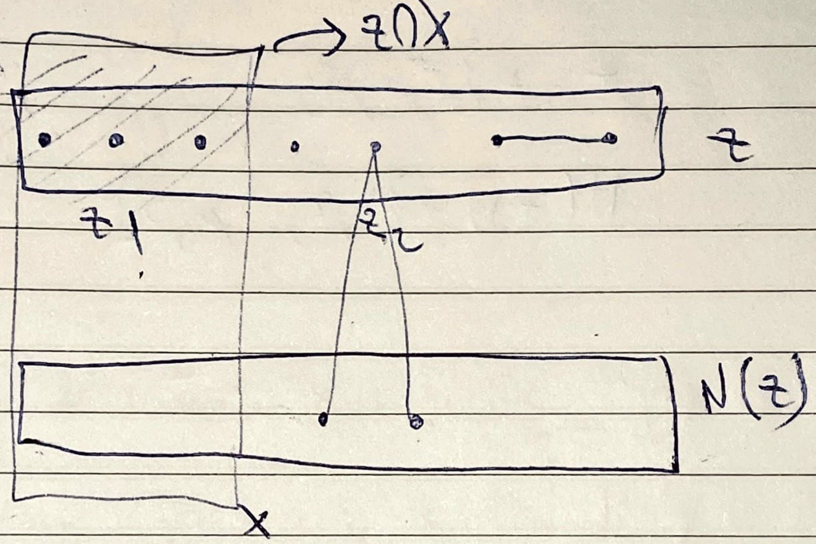
Fie Z sol. ret. de alg. de aprox.

Fie X sol. optimă

Fie $X \cap Z = Z_1$

$Z_2 = Z \setminus Z_1$

$X = Z_1 \cup N(Z_2)$



Algoritmul FPT

1) Rulez alg. de 2-aprox și găsesc o mț. Z

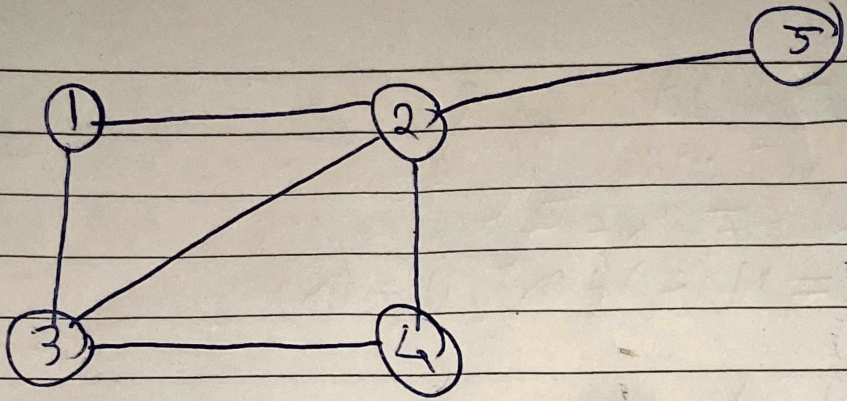
2) Dc. $Z > 2k \Rightarrow$ NU

3) Iterez după toate mț. $Z_1 \subseteq Z$ de ordinul ≤ 1

Dc. $|Z_1 \cup N(Z_2)| \geq k$, trec la pasul urm.

$Z_2 \Rightarrow \setminus Z_1$

Dc. $|Z_1 \cup N(Z_2)| \leq k \Rightarrow$ DA.



$Z = \{1,2,3,4\}$ $\quad$ $X = \{2,3\}$

$Z_1 = \{1\}$ × $\quad$ $K = 2$ $\quad$ $Z_1 = \{2,3\}$

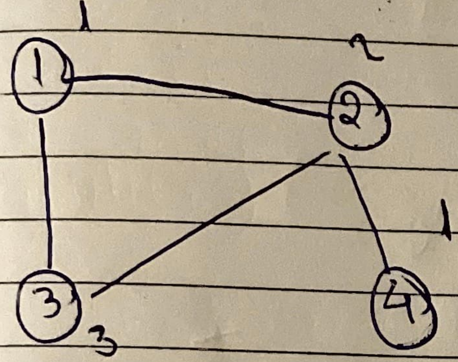
$Z_2 = \{2,3,4\}$ $\quad$ $Z_2 = \{1,4\}$

$N(Z_1) = \{1,2,3,4,5\}$ $\quad$ $Z_1 \cup N(Z_1) = \{2,3\}$ ✓

Seminarul 6

Prog. alg. ef.

Se dă un graf neorientat $G = (V, E)$ și un nr. k. Poate fi colorat G cu $n-k$ culori, unde $|V| = n$ culori distincte a.î. $(a,b) \in E$, $c(a) \neq c(b)$



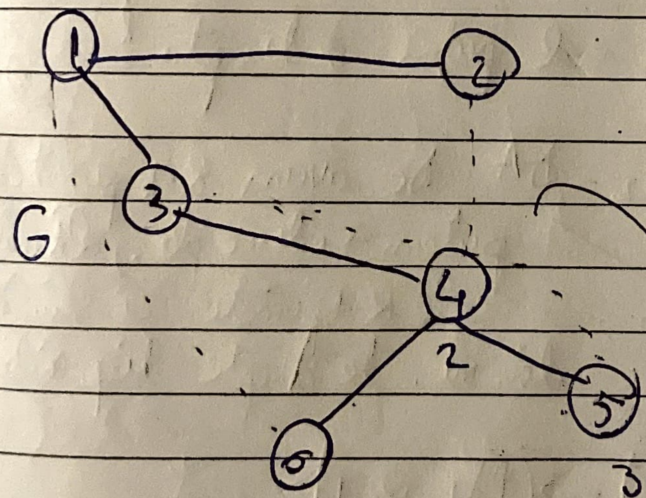
$K = 1$ DA

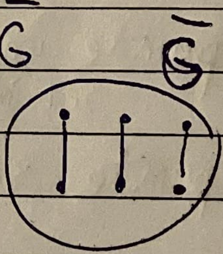
$K = 2$ NU

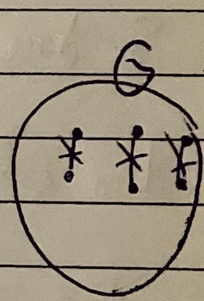
$\overline{G}$

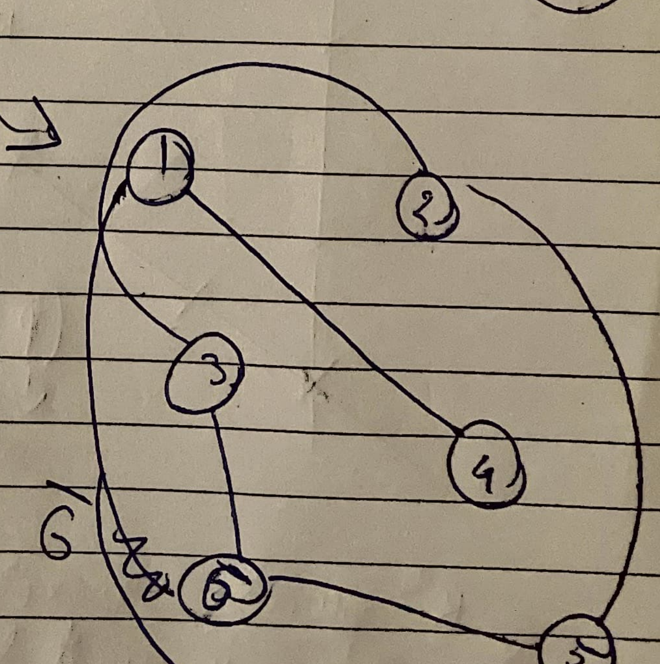
Lemai

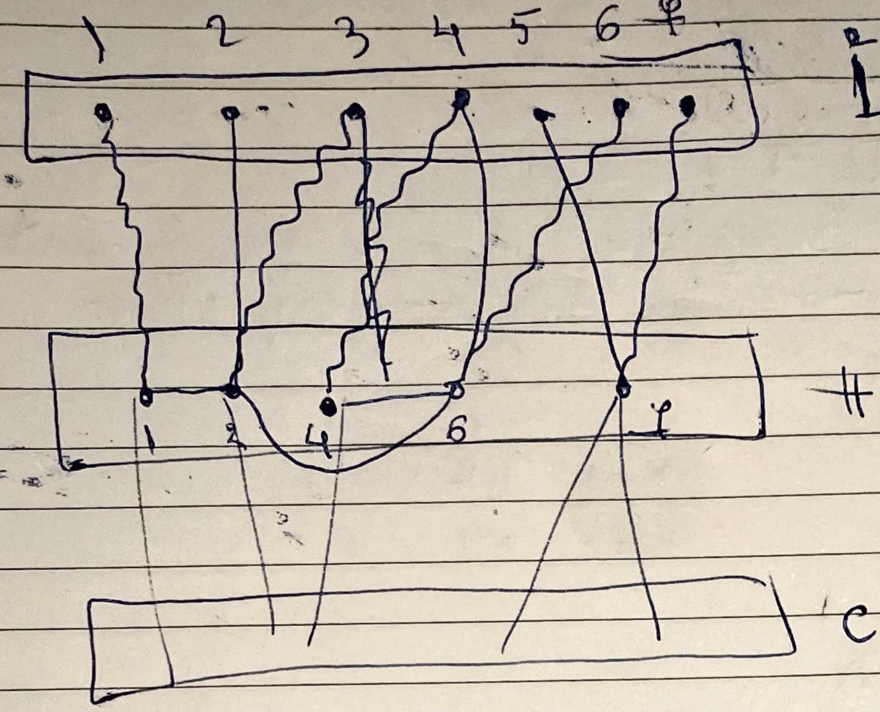
1) Avem un cuplaj de mărime $K+1$ în $\overline{G}$









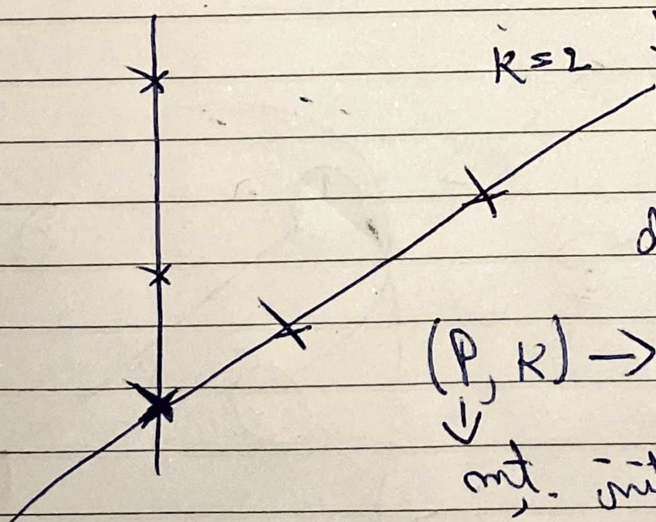


$\hat{I} \quad (G, k) \rightsquigarrow (G \setminus \{i \cup H\}, k - |H|)$

între i și C nu am muchie în $\vec{G}$

Se dau n puncte în plan. Pot ~~tr~~ trasa K drepte și K întregi a.î. fiecare din cele n puncte să fie pe cel puțin o dreaptă?

Ex:



K=2 DA Dc. avem K+1 puncte coliniare, putem reduce pb., eliminând cele K+1 puncte și scăzând 1 din nr. de drepte disponibile.

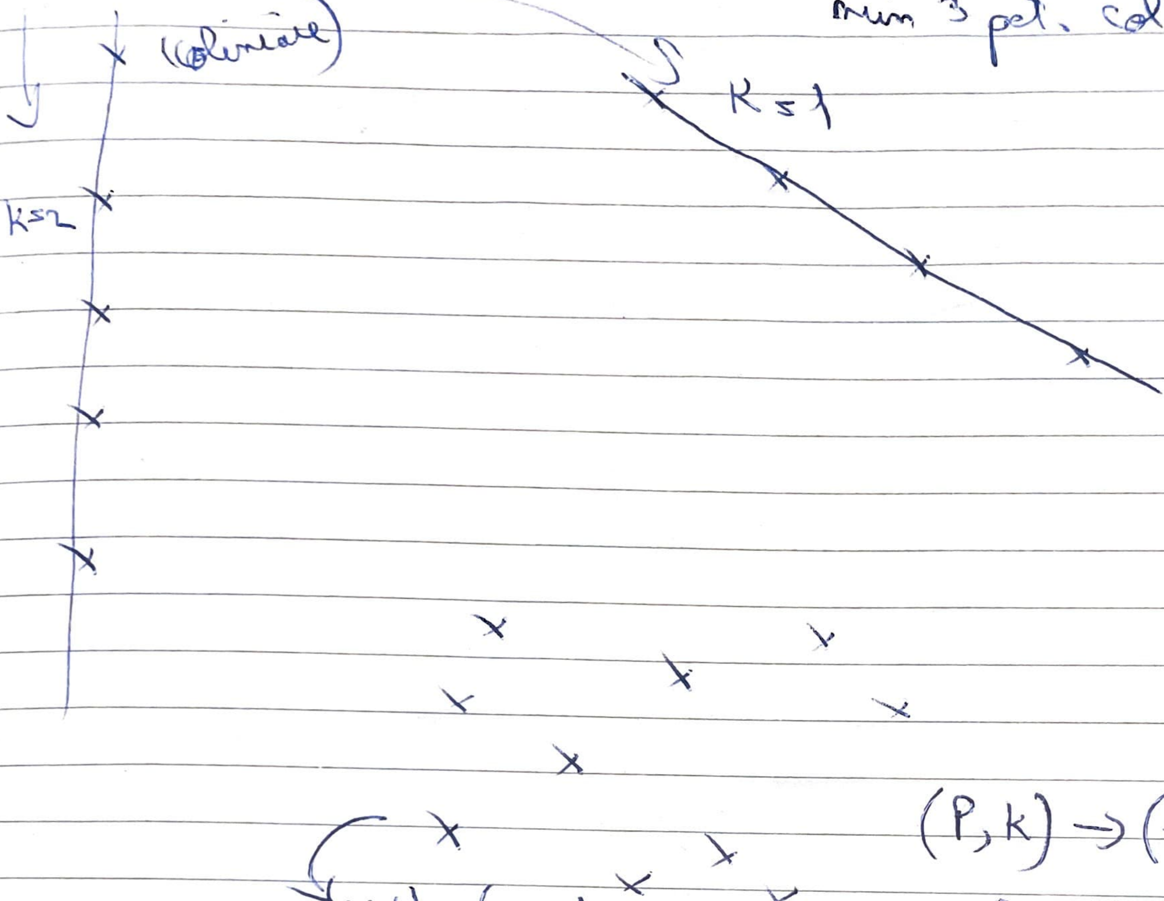
$(P, K) \to (P \setminus l, K-1)$

$\downarrow$ mt. inițială

K^2 puncte max în

Dc. am mai mult de K^2 puncte $\Rightarrow$ NU

$K=3$

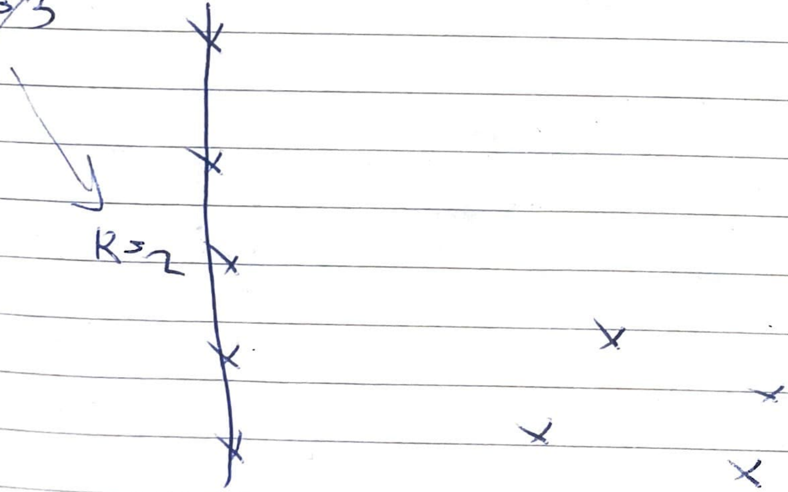


min 3 pct. col!!

$(P,k) \to (P \setminus i, k-1)$

$\Rightarrow$ NU (sunt mai mult de k^2 puncte

$K \geq 3$

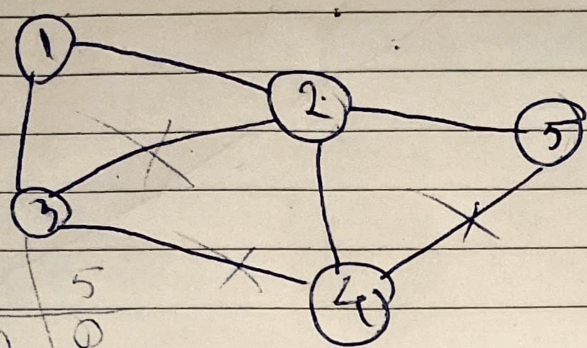


$C_k^{k^2}$

$(P,k) \to (P \setminus i, k-1)$

Se dă un graf neorientat $G = (V, E)$ și un nr. k. ~~Pot~~ Eliminăm k muchii a.î. G să nu aibă triunghiuri (cicluri de lung. 3)?

Ex:



~~K~~ $K = 1$ NU

$K = 2$ DA

	1	2	3	4	5
1	0	1	1	0	0
2	~~1~~	0	1	1	1
3	1	~~1~~	0	~~1~~	0
4	0	1	0	0	~~1~~ 0
5	0	1	0	0	0

ALG (G, K)

$\exists$ ciclu de lung. 3

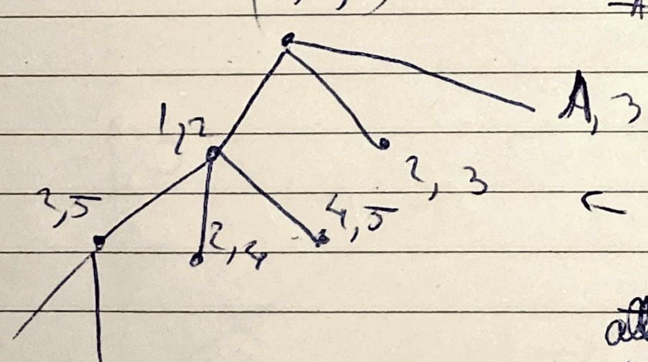
NU $\Rightarrow$ OK

DA $\quad k = 0 \Rightarrow$ nu se poate

altfel: ~~ALG $(G \setminus \{a, b\}, K-1)$;~~

ALG $(G \setminus \{b, c\}, K-1)$;

ALG $(G \setminus \{a, c\}, K-1)$;



$\leftarrow$ dc. pe ultima linie este NU $\Rightarrow$ OK

altfel $\Rightarrow$ am găsit ciclu de lung. k (am găsit minim o muchie de scos)

$O^*(3^k)$