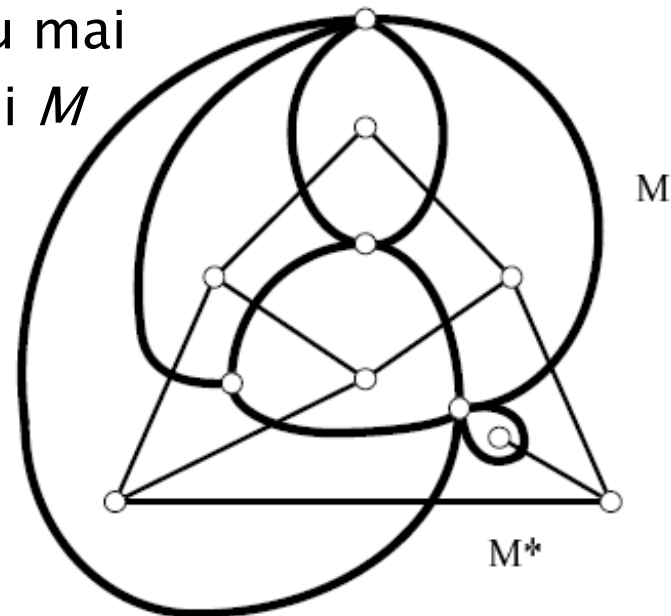


Graf planar

▶ Harta duală

- ▶ $M=(V,E,F) \rightarrow$ **harta duală** $M^*=(V^*, E^*, F^*)$
 - $V^* \Leftarrow$ se consideră câte un punct f^* în interiorul fiecărei fețe f din M
 - $E^* \Leftarrow$ pentru fiecare muchie e a lui M comună la două fețe f' și f'' se construiește un segment de curbă continuă e^* cu capetele în vârfurile f'^* și f''^* asociate fețelor f' și f'' astfel încât să intersecteze în interior muchia e și să nu mai intersecteze astfel nicio altă muchie a lui M



Graf planar

▶ Harta duală

▶ $M=(V,E,F) \rightarrow$ **harta duală** $M^*=(V^*, E^*, F^*)$

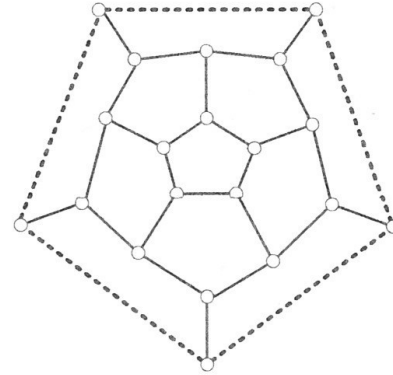
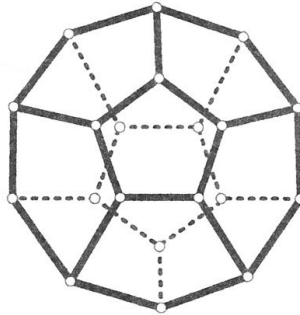
- $(M^*)^*$ este izomorf cu M

- $d_{M^*}(f^*) = d_M(f)$

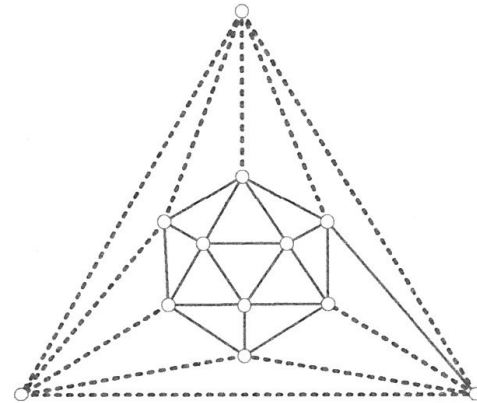
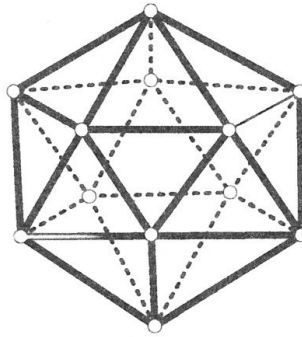
(gradul vârfului corespunzător feței în dual = gradul feței în M)

Graf planar

Dodecaedru:

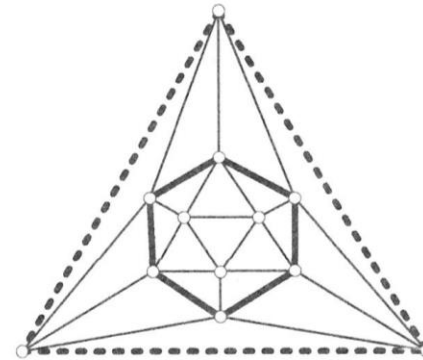
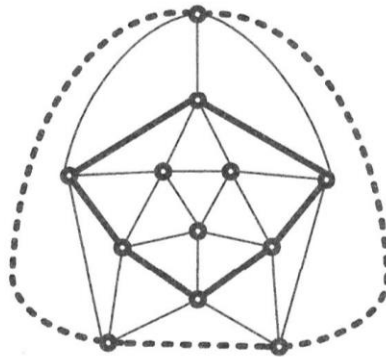
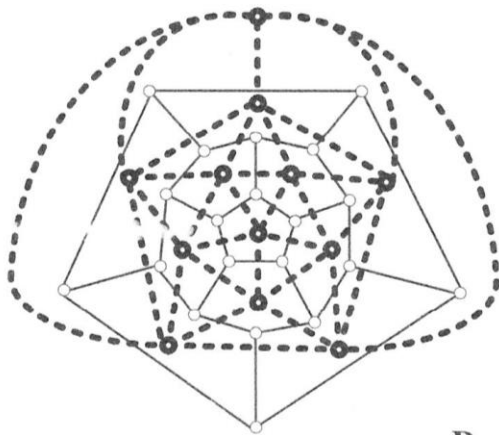


Icosaedru:

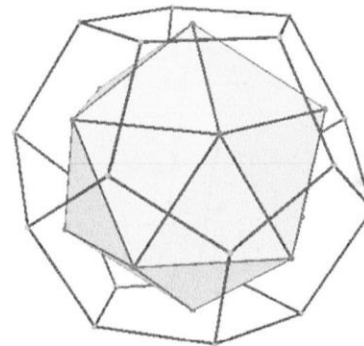
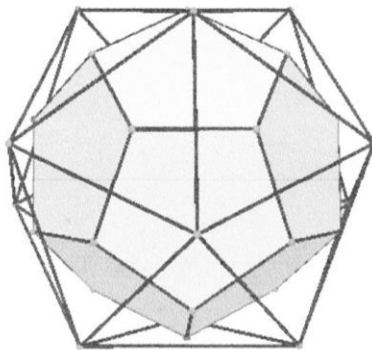


duale

Graf planar

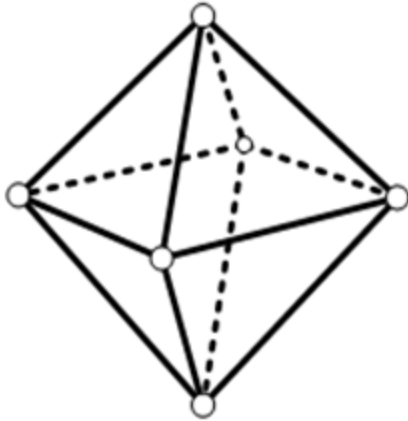


Dualul dodecaedrului = icosaedru

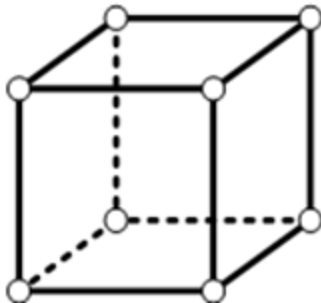
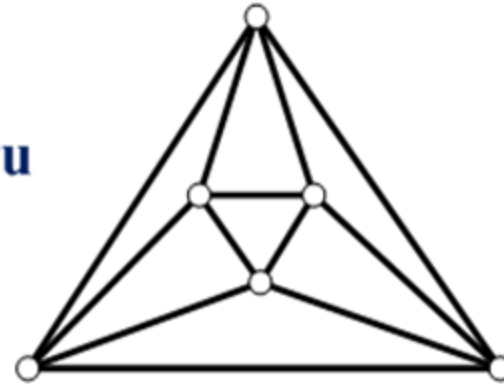


Dualitatea dodecaedru \Leftrightarrow icosaedru (<http://apollonius.math.nthu.edu.tw/d1/dg-07-exe/943251/dynamic/duality.htm>)

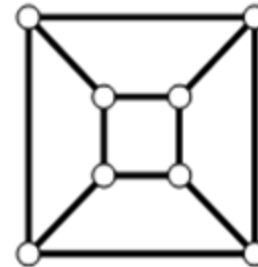
Graf planar



Octaedru

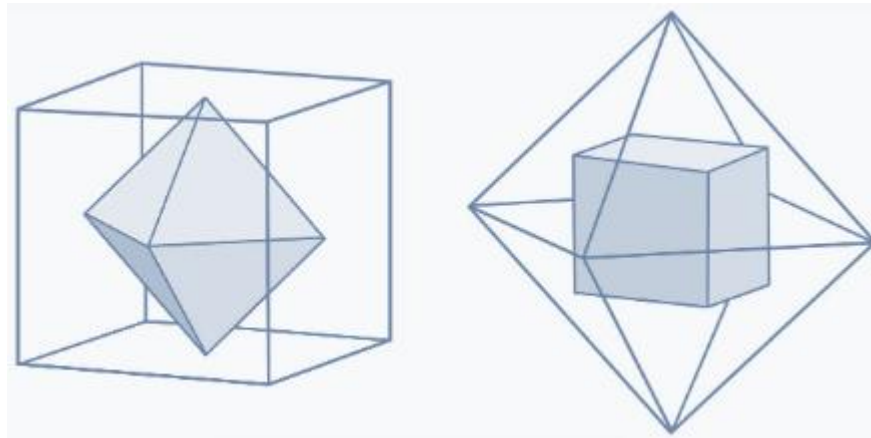
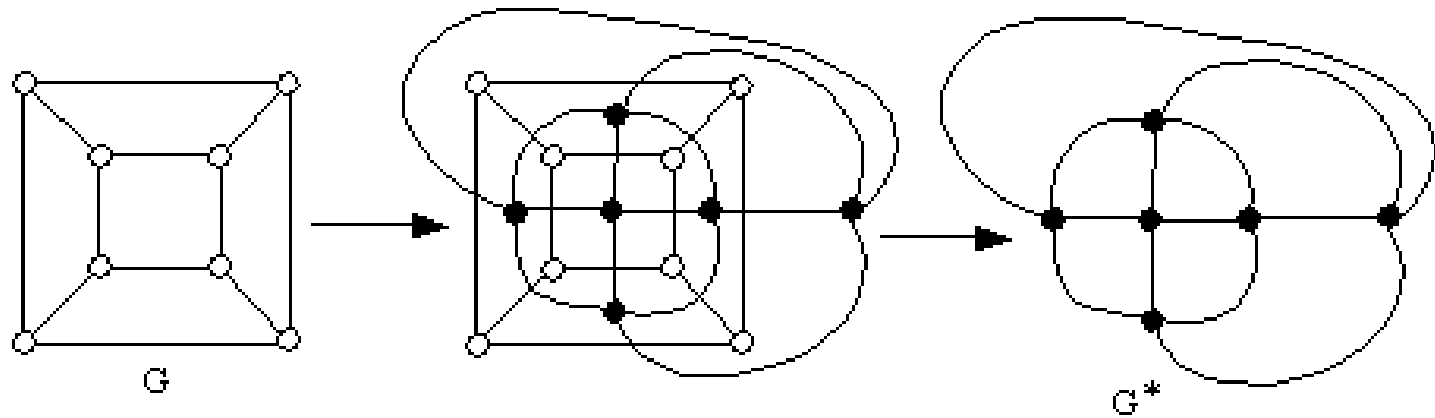


Cub

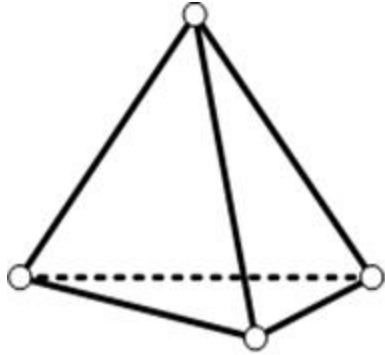


duale

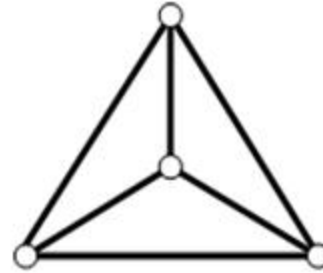
Graf planar



Graf planar



Tetraedru



Autodual (M izomorf cu M^*)

Colorări în grafuri



Colorări ale grafurilor

Amintim:

Compuțational: Dat p , este G p -colorabil?

Care este p minim cu proprietatea că G este p -colorabil?
= Care este numărul cromatic al lui G ?

- Este G 2-colorabil / graf bipartit – algoritm polinomial
- Este G 3-colorabil – problemă NP-completă

P/NP

- ▶ Complexitatea în timp a algoritmilor joacă un rol esențial.
- ▶ Un algoritm este considerat "acceptabil" numai dacă timpul său de executare este polinomial



Nu știm algoritm polinomial – problemă grea?

P/NP



Nu știm algoritm polinomial – problemă grea?

P = clasa problemelor pentru care există
algoritmi polinomiali (determiniști)

P/NP



Nu știm algoritm polinomial – problemă grea?

NP

- există algoritm polinomial pentru a testa o soluție candidat dacă este soluție posibilă (**verificator polinomial**) / există algoritm polinomial nedeterminist
- ⇒ o problemă NP poate fi rezolvată în timp exponențial (considerând toate soluțiile candidat)

P/NP



Nu știm algoritm polinomial – problemă grea?

NP

- $P \neq NP$?
- Probleme NP-complete
 - $B \in \text{NP}$ a.î. $\forall A \in NP, A \leq_p B$ (reducere în timp polinomial)
 - Dacă pentru un B se găsește algoritm polinomial, atunci $P = NP$
 - SAT (Cook–Levin)
- Probleme NP-dificile (NP-hard)
 - B a.î. $\forall A \in NP, A \leq_p B$.

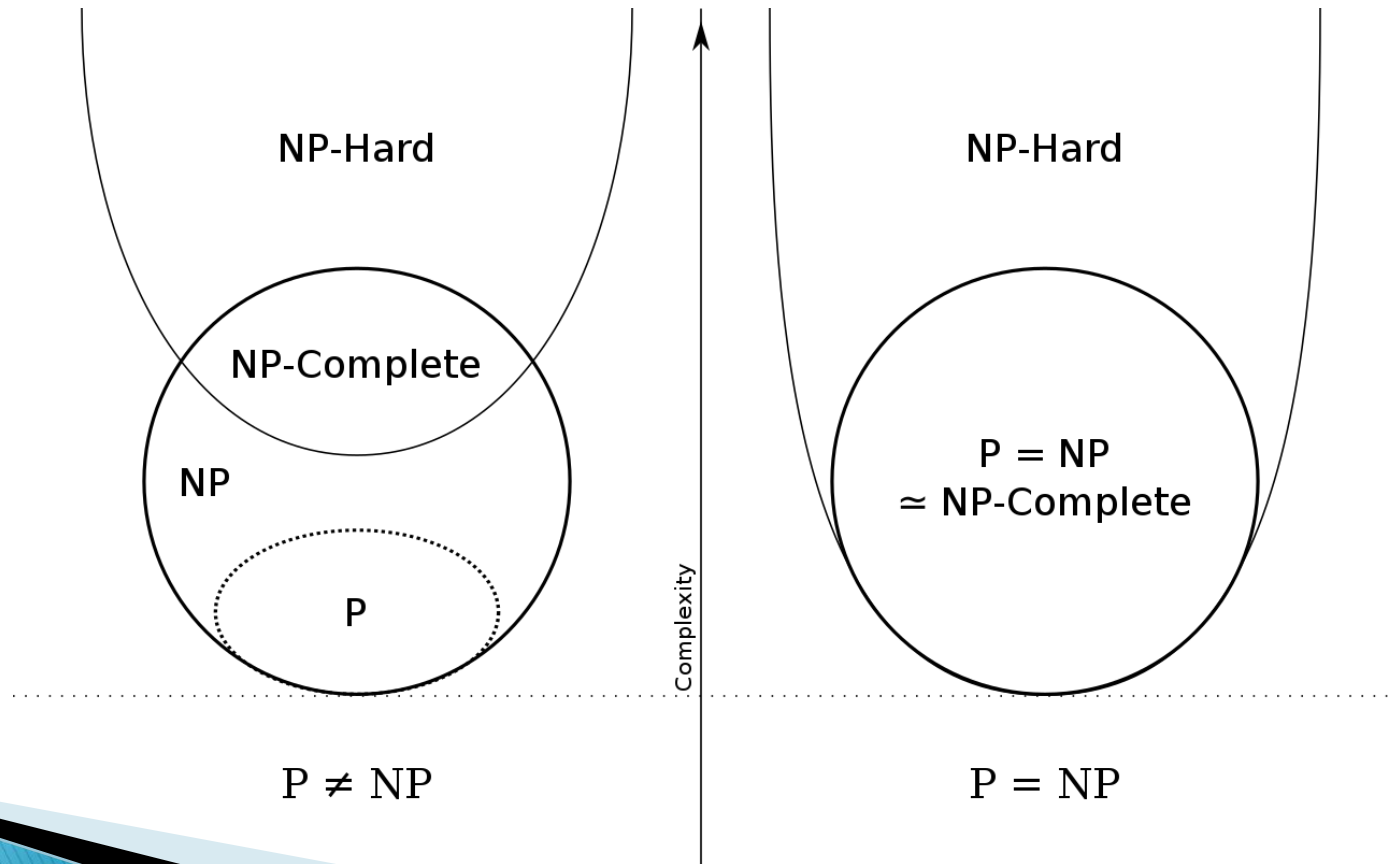
P/NP



Nu știm algoritm polinomial – problemă grea?

NP

– P versus NP



Metode de elaborare a algoritmilor



Nu știm algoritm polinomial

- Demonstrăm NP – dificilă

Soluții:

- algoritmi exponențiali mai rapizi decât cei exhaustivi (brute force) de căutare în spațiul soluțiilor: **Backtracking, Branch & Bound**
- **Compromis:** algoritmi mai rapizi care produc soluții care nu sunt optime – algoritmi euristici, aleatorii, genetici...

Algoritmi de colorare de tip greedy (euristici)

Colorări în grafuri

Algoritm Greedy de colorare

- ▶ Fie v_1, \dots, v_n o ordonare a vârfurilor
- ▶ Pentru $i = 1, \dots, n$
 - Colorează v_i cu cea mai mică culoare posibilă (care nu este culoare a unui vecin al său deja colorat)

Colorări în grafuri

Algoritm Greedy de colorare

► Complexitate?

- $O(n+m)$ – determinarea primei culori disponibile pentru v – ordin $d(v)$

Colorări în grafuri

Algoritm Greedy de colorare

- ▶ Câte culori folosește maxim? (\Rightarrow limită superioară pentru numărul cromatic)
- ▶ Cât de mare poate fi diferența între numărul de culori folosite de Algoritmul Greedy de colorare și numărul cromatic? Sunt clase de grafuri pentru care avem egalitate?

Colorări în grafuri

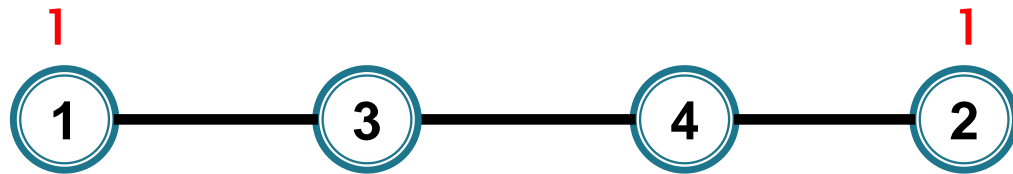
Algorithm Greedy de colorare

- ▶ Ordonarea contează, în funcție de ea numărul de culori poate diferi

Exemplul 1:



Ordinea 1, 2, 3, 4:



Colorări în grafuri

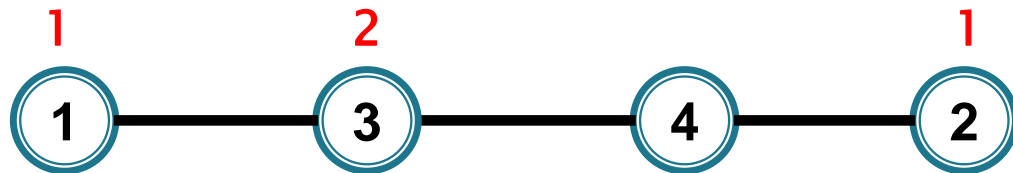
Algorithm Greedy de colorare

- ▶ Ordonarea contează, în funcție de ea numărul de culori poate diferi

Exemplul 1:



Ordinea 1, 2, 3, 4:



Colorări în grafuri

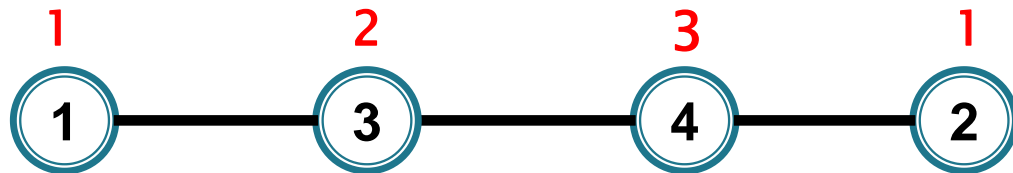
Algorithm Greedy de colorare

- ▶ Ordonarea contează, în funcție de ea numărul de culori poate diferi

Exemplul 1:



Ordinea 1, 2, 3, 4:



Colorări în grafuri

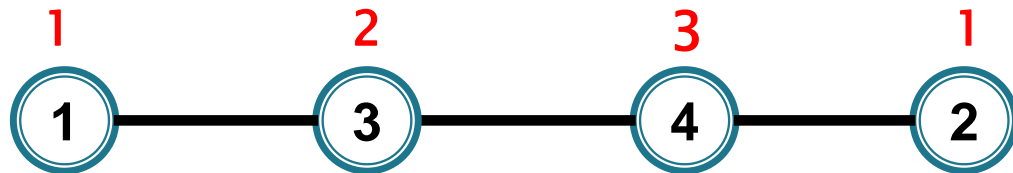
Algoritm Greedy de colorare

- ▶ Ordonarea contează, în funcție de ea numărul de culori poate diferi

Exemplul 1:



Ordinea 1, 2, 3, 4:



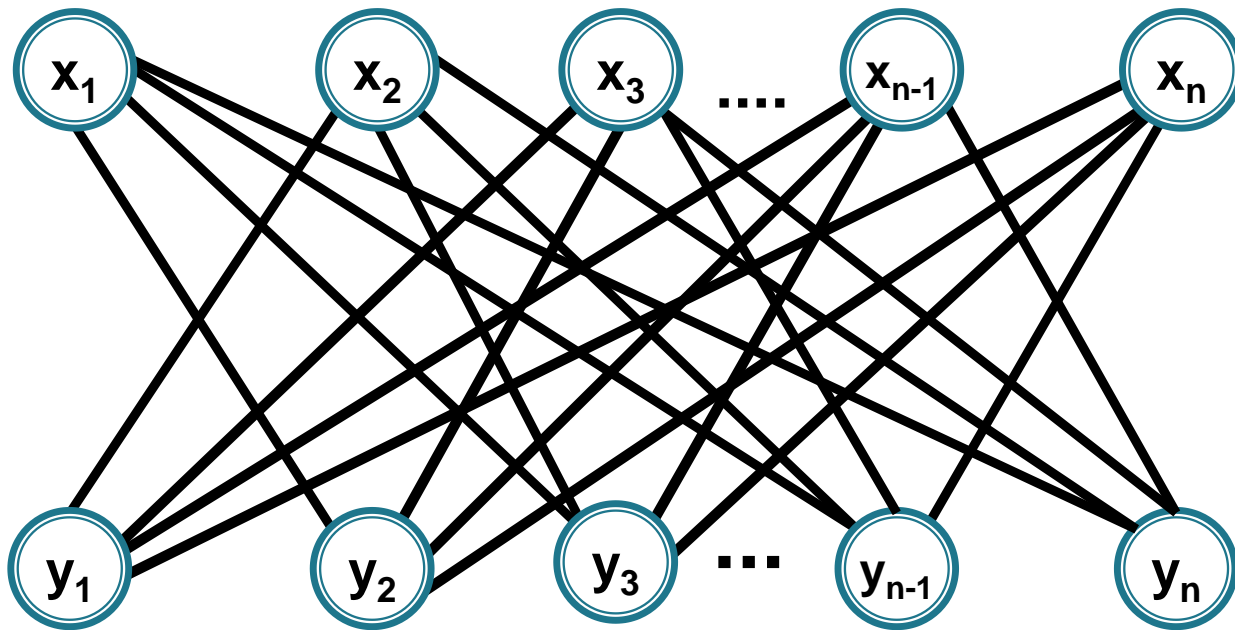
Ordinea 1, 3, 4, 2:



Colorări în grafuri

Algoritm Greedy de colorare

Exemplul 2: – Graful $G = K_{n,n}$ fără muchiile $x_i y_i$



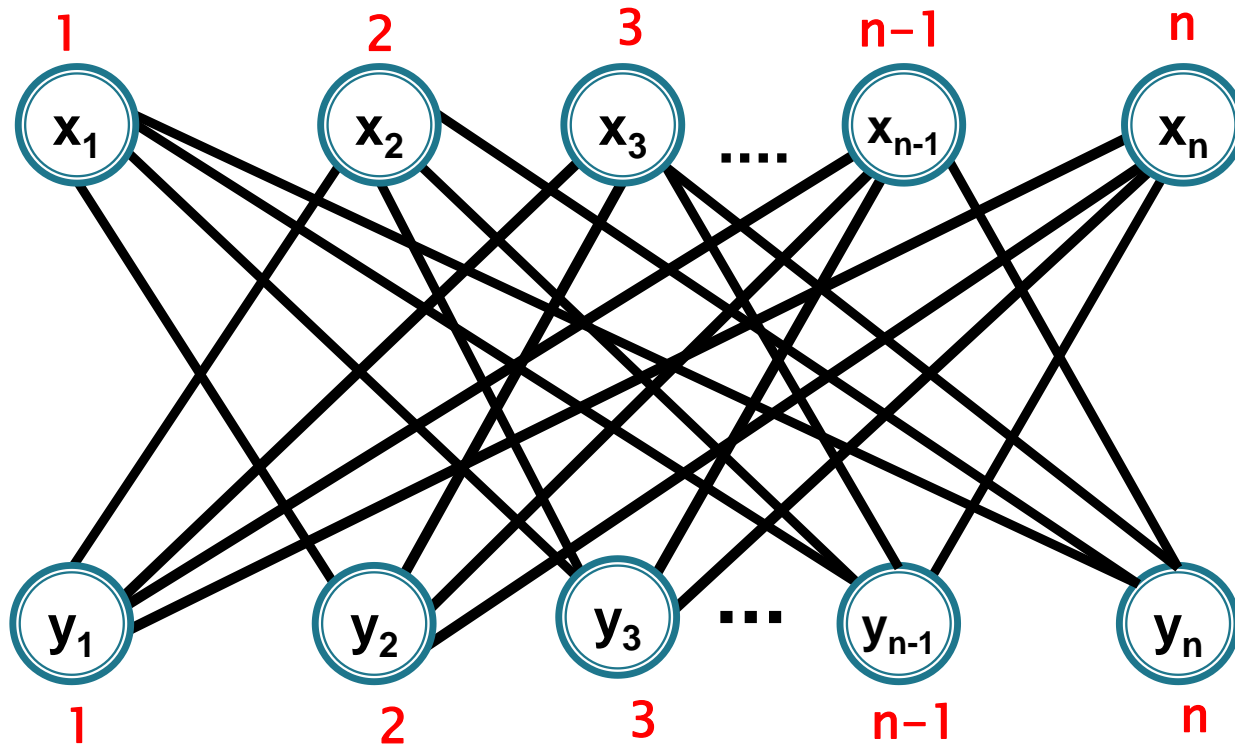
Ordinea $x_1, y_1, x_2, y_2, x_3, y_3, \dots, x_n, y_n$: ? culori

$$\chi(G) = 2$$

Colorări în grafuri

Algoritm Greedy de colorare

Exemplul 2: – Graful $G = K_{n,n}$ fără muchiile $x_i y_i$



Ordinea $x_1, y_1, x_2, y_2, x_3, y_3, \dots, x_n, y_n$: $n = \frac{|V(G)|}{2}$ culori

$$\chi(G) = 2$$

Colorări în grafuri

Algoritm Greedy de colorare

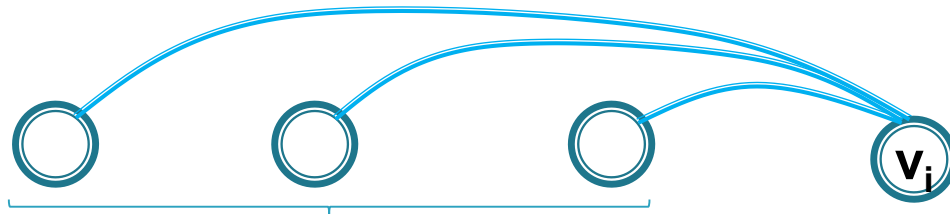
- ▶ $N_{\text{greedy}}(G, \{v_1, \dots, v_n\})$ = numărul de culori folosit de Algoritmul Greedy de colorare pentru G considerând vârfurile în ordinea v_1, \dots, v_n
- ▶ $\Delta(G)$ = gradul maxim al lui G

Colorări în grafuri

Algorithm Greedy de colorare

▶ $N_{\text{greedy}}(G, \{v_1, \dots, v_n\}) \leq \max\{ \min(d(v_i) + 1, i) \mid i = 1, \dots, n \}$

$$\leq \Delta(G) + 1$$



v_i are $d(v_i)$ vecini și sunt cel mult $i-1$ vârfuri deja colorate \Rightarrow
 v_i are cel mult $\min\{d(v_i), i-1\} \leq \Delta(G)$ vecini deja colorați
când se alege culoare lui

Colorări în grafuri

Limite pentru numărul cromatic

Proprietate

$$\chi(G) \leq \Delta(G) + 1$$

(Demonstrație: $\chi(G) \leq N_{\text{greedy}}(G, \{v_1, \dots, v_n\})$)

Observații:

1. $\chi(K_n) = n = \Delta(K_n) + 1$
2. $\chi(C_n) = 3 = \Delta(C_n) + 1$ pentru n impar
3. $\chi(S_n) = 2$, $\Delta(S_n) = n - 1$ (graful stea)

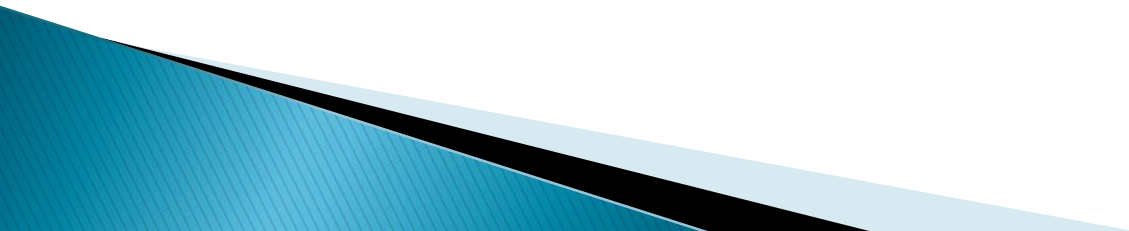
Colorări în grafuri

Limite pentru numărul cromatic

Teorema lui Brooks

$\chi(G) \leq \Delta(G)$ pentru G care nu este graf complet
sau ciclu impar

(Demonstrație– suplimentar, v. bibliografie)



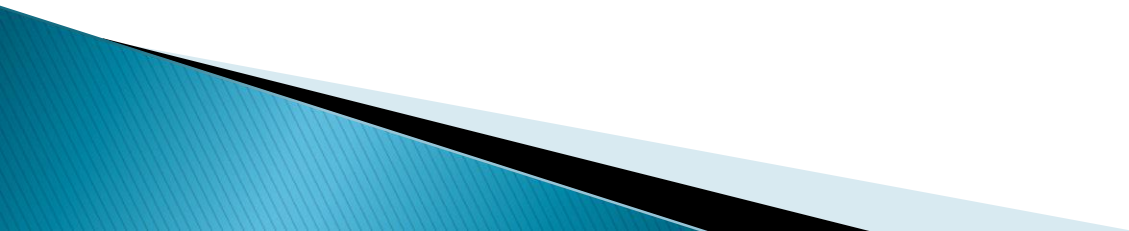
Colorări în grafuri

Limite pentru numărul cromatic

Proprietate

$\chi(G) \geq \omega(G)$ = numărul clică = cardinalul maxim al
unei clici (subgraf complet) din G

Demonstrație – vârfurile dintr-o clică trebuie
demonstrate diferit



Colorări în grafuri

Algoritm Greedy de colorare

- ▶ Am demonstrat la grafuri planare: există o ordonare a vârfurilor pentru care Algoritmul Greedy furnizează o colorare cu cel mult 6 culori (Teorema celor 6 culori)
- ▶ Exista ordonări + clase de grafuri pentru care Algoritmul Greedy de colorare este optim?

Colorări în grafuri

Algoritm Greedy de colorare

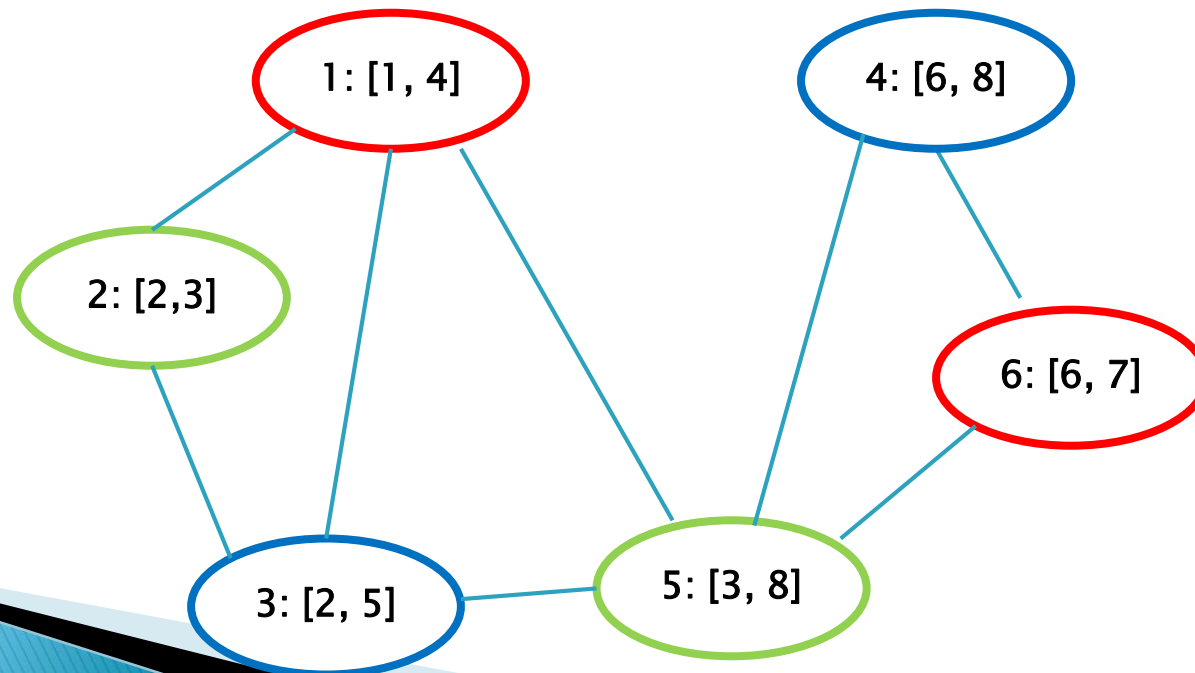
- ▶ Exista ordonări + clase de grafuri pentru care Algoritmul Greedy de colorare este optim?
 - Grafuri bipartite, ordonare dată de o parcurgere (sau orice ordonare în care orice vârf v_i este adiacent cu cel puțin un vârf v_j cu $j < i$)
 - Grafuri interval

Colorări în grafuri

Algoritm Greedy de colorare

► Grafuri interval

- Fiecare vârf i – asociat unui interval $[a_i, b_i]$
- Muchii – între intervale care se intersectează



Colorări în grafuri

Algoritm Greedy de colorare

► Grafuri interval

Proprietate: Fie G un graf interval si v_1, \dots, v_n o ordonare a vârfurilor sale după extremitatea inițială a intervalelor corespunzătoare.

Avem $\chi(G) = N_{\text{greedy}}(G, \{v_1, \dots, v_n\})$

Demonstrație – tablă

Colorări în grafuri

Algoritm Greedy de colorare

▶ Grafuri interval

Algoritm $O(n \log(n))$ de determinare a numărului cromatic $\chi(G)$ și a unei $\chi(G)$ -colorări – vezi metoda Greedy, problema Partiționării intervalelor

Colorări în grafuri

Agoritm Greedy de colorare

Ordonări ale vârfurilor – strategii generale

- ▶ **SL Smallest Last** [Matula et al]: v_1, \dots, v_n astfel încât v_i este vârful de grad minim din $G - v_n - \dots - v_{i+1}$
 - folosește cel mult 6 culori pentru grafuri planare
- ▶ **LF Largest first** [Welsh, Powell]: v_1, \dots, v_n în ordine descrescătoare după grad

Colorări în grafuri

Agoritm Greedy de culoare

- ▶ Ordonări dinamice:
 - DSatur – se dă prioritate în ordonare vârfurilor care au un număr maxim de vecini deja colorați (și, în caz de egalitate, celor cu gradul cel mai mare)
 - optim pentru grafuri bipartite

Colorări în grafuri

Agoritm Greedy de culoare

- ▶ Repetarea algoritmului pe ordonări diferite:

repetă în timpul avut la dispoziție:

- generează aleator o ordonare a vârfurilor
 - colorează G folosind algoritmul Greedy pentru această ordonare
 - dacă colorarea obținută folosește un număr mai mic de culori decât cea mai bună găsită până acum, memorează această colorare ca fiind cea mai bună
- 