

DRZEWA ROZPINAJĄCE

1. W pełnym grafie K_n (mającym n wierzchołków i wszystkie krawędzie) jest n^{n-2} drzew rozpinających. Przypomnijmy, że kod Prüfera wyznacza bijekcję pomiędzy zbiorem drzew rozpinających i zbiorem $C = \{1, \dots, n\}^{n-2}$, działając według recepty

Dla grafu mającego co najmniej 2 krawędzie zapisz numer wierzchołka, do którego przyłączony jest liść o najmniejszym numerze; następnie usuń ten liść.

Zauważyć, że różne drzewa rozpinające mają różne kody. Następnie trzeba wykoncypować metodę rysowania drzewa na podstawie kodu (lub zajrzeć do [Wiki](#)). To pokaże, że każdy ciąg ze zbioru C koduje pewne drzewo rozpinające.

2. *Grafem dualnym* do grafu planarnego (V, E) (wyznaczającego zbiór ścian F) nazwiemy graf (F, E) , w którym każda krawędź łączy obszary, które rozdziela.

Inaczej mówiąc, można utożsamić ścianę $f \in F$ z pewnym punktem p_f do niej należącym i połączyć krawędziami te pary p_f i p_g , dla których ściany f i g mają wspólną krawędź w wyjściowym grafie.

Pokazać, że jeśli (V, E') jest drzewem rozpinającym w planarnym grafie (V, E) to $(F, E \setminus E')$ jest drzewem rozpinającym w grafie dualnym (F, E) . Wywnioskować stąd formułę Eulera.

SYSTEMY RÓŻNYCH REPREZENTANTÓW (SRR)

Przypomnijmy, że ciąg zbiorów $\mathcal{A} = (A_1, A_2, \dots, A_n)$ ma system różnych reprezentantów (SRR), zwany też transwersalą, jeżeli istnieją **różne** punkty $x_1, \dots, x_n$, takie że $x_i \in A_i$ dla każdego $i \leq n$.

3. Niech $\mathcal{A} = (A_1, A_2, \dots, A_n)$ będzie ciągiem zbiorów spełniającym silniejszy Warunek Halla

$$|A_{i_1} \cup A_{i_2} \cup \dots \cup A_{i_k}| \geq k + 1,$$

dla każdego $k = 1, 2, \dots, n$ i dowolnego wyboru k różnych indeksów $i_1, i_2, \dots, i_k$. Niech x będzie elementem z A_1 . Pokazać, że $\mathcal{A}$ ma SRR, w którym x reprezentuje A_1 .

4. Udowodnić, że jeżeli każdy zbiór z ciągu $\mathcal{A}$ ma $\geq d$ elementów, a każdy punkt należy do co najwyżej d zbiorów z $\mathcal{A}$ (gdzie $d > 0$ jest ustalone) to taki ciąg ma SRR.
5. Udowodnić, że ciąg $\mathcal{A} = (A_1, A_2, \dots, A_n)$ niepustych podzbiorów skończonego zbioru X ma SRR wtedy i tylko wtedy gdy $|\{i : A_i \subseteq Y\}| \leq |Y|$ dla każdego $Y \subseteq X$.
6. Po rozgrywce brydżowej talia 52 kart jest podzielona na 13 lew po 4 karty. Udowodnić, że można z każdej lewy wybrać po jednej karcie, tak aby otrzymać wszystkie figury 2, 3, $\dots$, 10, W, D, K, A (bez uwzględniania kolorów).

KOJARZENIE PAR

Przypomnijmy, że w danym grafie dwudzielnym $G = (S, T, E)$, jeżeli $M \subseteq E$ jest skojarzeniem par to łańcuch alternujący pozwala skojarzyć $|M| + 1$ par. Poszukiwanie łańcucha alternującego przebiega następująco:

- (i) nadaj etykietę (*) wszystkim tym wierzchołkom z S , które nie mają pary w M ;
- (ii) jeśli $s_i \in S$ dostał w poprzednim ruchu etykietę, nadaj etykietę (s_i) tym wierzchołkom z T , które są połączone krawędzią **spoza** M i dotychczas nie miały etykiety;
- (iii) jeśli $t_i \in T$ dostał w poprzednim ruchu etykietę, nadaj etykietę (t_i) tym wierzchołkom z S , które są połączone krawędzią **z** M i nie miały etykiety.

Każdy wierzchołek może dostać co najwyżej jedną etykietę. Jeżeli etykietę dostał wolny wierzchołek z T to w ten sposób znajdujemy łańcuch alternujący. W przeciwnym przypadku, jak udowodnimy na wykładzie, skojarzenie jest maksymalne. Warto poćwiczyć!

7. Zauważyć, że twierdzenie Halla o małżeństwach i twierdzenie o SRR są równoważne w tym sensie, że jedno łatwo wynika z drugiego.
8. Zauważyć, że jeśli w grafie dwudzielnym $G = (S, T, E)$ stopień każdego wierzchołka z S jest dodatni i nie mniejszy niż stopień dowolnego wierzchołka z T to w G można skojarzyć $|S|$ par.
9. Zauważyć, że jeżeli w grafie dwudzielnym $G = (S, T, E)$, gdzie $|S| = |T| = n$ nie ma zbiorów blokujących mocy $< n$ to graf spełnia warunek Halla $|G[A]| \geq |A|$ dla każdego $A \subseteq S$.
10. Wyprowadzić twierdzenie Halla o małżeństwach z twierdzenia Königa-Egerváry'ego:

W grafie dwudzielnym maksymalna liczba skojarzonych par jest równa minimalnej mocy zbioru blokującego.