МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЦЕНТРАЛЬНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Механіко-технологічний факультет

Кафедра кібербезпеки та програмного забезпечення

ДИСКРЕТНА МАТЕМАТИКА

Методичні вказівки

для студентів заочної форми навчання за спеціальностями 123 “Комп’ютерна інженерія”

ЗАТВЕРДЖЕНО

на засіданні кафедри кафедри кібербезпеки та програмного забезпечення, протокол від 28 березня 2018 року № 14

КРОПИВНИЦЬКИЙ

2018

Дискретна математика: метод. вказівки для студ. заочної форми навч. за спец. 123 “Комп’ютерна інженерія” / уклад. Петренюк В.І. — Кропивницький: ЦНТУ, 2018. — 77 с.

Укладач: Петренюк В.І., канд. фіз.-мат. наук, доцент

Рецензент: Волков Ю. І., д-р фіз.-мат. наук, професор;

Схвалено на засіданні методичного семінару
кафедри кібербезпеки та програмного забезпечення
(протокол від 05 липня 2017 року № 1)

Вступ до дискретної математики

Дискретна математика – це наука, що вивчає математичні властивості дискретних абстрактних структур.

Складається з комбінованого аналізу:

-комбінаторного аналізу,

-теорії графів,

- математичної логіки.

Нескінченності вітсутнє як нескінченно малі так і нескінченно великі, а границі визначаються на основі наявності стійкості.

Перехід від нескінченності до дискретності (величини)

$x: = x + \alpha \uparrow_{числа} \to x := x + \alpha \uparrow_{числа}$

Границею нескінченної числової непослідовності $\{x_n, 1_{n-1}, x_1, x_2, x_3\}$ називатимемо число А, яке задовольняє нерівності $|x_n - A| < \epsilon$ для всіх n, $n \geq N$ та довільного наперед заданного числа (епсілон).

Нескінченна матеріальна величина має своєю границею нуль, а нескінченна величина – безкінечність.

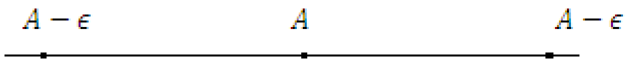
$\lim \quad \alpha = \emptyset - \alpha$ Нескінченна мала величина.

$\lim \quad \beta = \infty - \beta$ Нескінченно велика величина.

На відміну від поняття нескінченної числової послідовності, що дається в тиці ми дискретизуемо ці поняття.

Для цього введемо поняття стійкості значень числової (нескінченоої) непослідовності, яке полягає в тему, що починаючи з якогось номера N всі члени числової непослідовності «скупчюються» навколо А. Тобто потрапляють в проміжок $[A - \epsilon, A + \epsilon]$

Тільки ϵ- фіксоване число



Задача №1

Скласти алгоритм виявлення стійкості заданої числової послідовності.

Задача №2

Визначити ймовірність появи в українському тексті (файлі)

1) перші літери вашого прізвища
2) першого складу вашого прізвища

1)»0» , «0» $\approx$ 0.58 (ймовірність)

2)

Практичне завдання №20

Обчислення ймовірності події присвячене розв'язання задачі №2, для «0» ш «о»

1)беремо текст із частини газетної статті

2)Складаємо таблицю для порції на 100 літер

100 літер

№порції	1	2	3		47	48	49	50
Кількість «0» ш «о»	4	5	6	$\epsilon = 0{,}002$	Числа без зростання			
частота	$\frac{4}{100}$	$\frac{4+5}{100}$	$\frac{4+5+6}{300}$	A=0,58	A_{47}	A_{48}	A_{49}	A_{50}

3)Висновок щодо стійкості робимо розглядаючи числа (50,49,48,47). Якщо в кінці таблиці(50,29,28..) є ближчими до числа A і відзначаються від цього (A) на одну тисячну.

За імовірність появи літери візьмемо середнє арифметичне цих чотирьох чисел, на якій спостерігається стійкість.

Хід розв'язання : задачі 1 для

1) Відкриваемо файл, β
2) Поки не кінець файла β виконувати:
3) Збільшуємо символ m
4) Якшо $m_1 = m(0), m = 0$, то збільшуємо

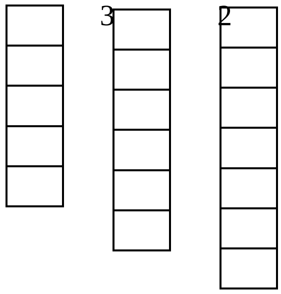
Основний принцип комбінаторики

Незалежність події висловлюємо життєвим досвідом або спираючись на формули P(AB)=P(A)*P(B).

Задача №2

Підрахувати кількість числових двоцифрових номерів, де друга цифра більша за першу цифру номера.

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



Відповідь 36 всього варіантів.

Того що подія вибору другої цифри таких номерів залежить від події вибору першої цифри (основний принцип тут не тпрацює). В такому випадку потріно перебрати всі варіанти використовуючи досвід або здоровий глузд.

Якшо потрібно строго математично довести залежність між двома подіями , то потрібно обчислити наведеним на першій лекції способ наявність події AB. А також потім обчислити ймовірності окремих подій AB

$$C_n^R = \frac{n!}{R!\,(n-R)!} = \frac{n(n-1)(n-2)(n-3)\ldots 2*1}{R(R-1)\ldots 2*1*(n-R)(n-R-1)\ldots *1}$$

C_n^R- кількість варіантів вибору

R елементів із множини з елементів

Задача

R=k=2, n=3

$$C_k^n = \frac{n!}{k!(n-(k))!}$$

$\{1,2,3\}$ - множина, з якої вибираємо по 2 елемента

$\{R_1, R_2\}$, вважаючи з «життевого досвіду», що вибір першого елемента не залежить від вибору 2 другого

$n_1 = 3\ ,n = (3-1) = 2$

$n_1 * n_2 = 3 * 2 = 6$

Перебираємо вручну

$$\left.\begin{matrix} 1 & 2 \\ 1 & 3 \end{matrix}\right| \begin{matrix} 2 & 1 \\ 2 & 3 \end{matrix} \left| \begin{matrix} 3 & 1 \\ 3 & 2 \end{matrix} \right\}$$ 6 варіантів вибору

Висновок: про порядок слідування елементів в множині з R елементів ,де R=2, n=3

Наведена вище формула C_n^R не враховує порядок слідування елементів в множині з n- елементів

«Про графи та їхній синтез»

Під графом G розумітимемо впорядковану пару множин

$G = G^0 G^{11}$ або $G = (V, E),$ де Vабо G^0множина вершин

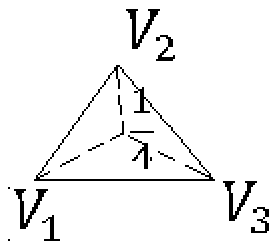
а G^1- множина ребер, що з'єднують вершини,

та діаграму (рисунок) на площині.

G=(V,E)= K_4- повний графічний порядок G

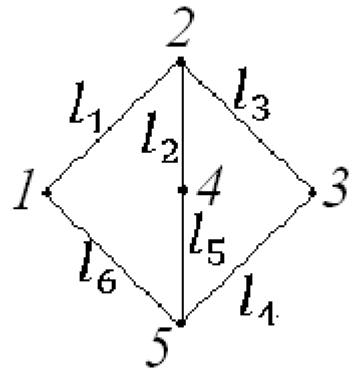
$V = \{V_1, V_2, V_3, V_4,\}$

$E = \{(V_1 V_2), (V_2 V_3), (V_3 V_4), (V_3 V_1), (V_1 V_4), (V_2 V_4)\}$



Граф $K_{2,3}$; G=(V,E)

V= {1,2,3,4,5}

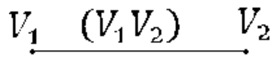


Обидва графи в цих прикладах певним чином синтезовані з трикутників та чотирикутників.

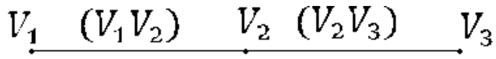
Будемо вважати, що графи складаються з двух основних елементів ланцюгів та циклів, з'єднаних між собою шляхом склеювання по вершинах (ребрам, або їх комбінаціями)

Під ланцюгомL будемо розуміти абстрактний шлях (по ребрах як по вулицям)

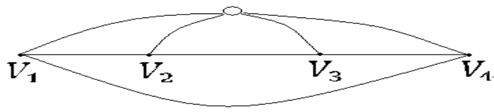
Під ланцюго розумітимемо послідовність ребер та вершини : $V_1\{V_1, V_2\}$

a) 

-Ланцюг довжини 1 графа K_4

b) 

друга графа K_4в ребро V_1, V_3, що з'єднує початкові та кінцеві вершини. Це цикл довжини 3.

c) 

= K_4

На цій діаграмі (b) пристрій перетин двох ліній в одній спільній внутріщній точці.

Під циклом графа розумітимемо замкнутий ланцюг (кінцеваV_n співпала з V_1) довжини ($V_1 = V_n$)

L=1,l_1,2,$2l_2$,4l_5,S- довжина 3

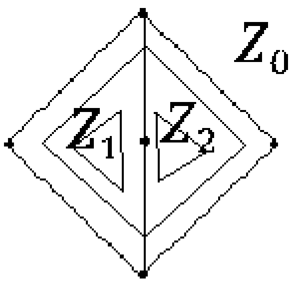
Простий ланцюг не має діагоналі.

Простий ланцюг називається такий ланцюг , в якому неможливо скоротити його довжину за допомогою ребер – містків(не скорочюваний ребер ланцюг).

Під синтезом графів розумітимемо операцію створення більших (повних) графів із менших графів шляхом ототожню вання деяких ланцюгів і циклів, метою буде «наслідування» новим графом повної властивості від менших графів.

Властивість ця є числом досяжності множини вершини, що дорівнює кількості граней (найменшому числу), на границях яких розміщено всі вершини графа.

Приклад 3 діаграма $K_{2,3}$розкізана на грані Z_i

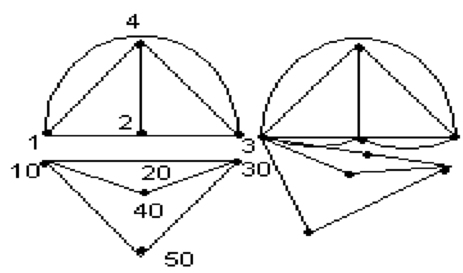


Якшо розрізати по ребрам , то з площини утворяться три грані.

I=0,1,2 Всі ці поверхні обмежені циклами довжини чотири (є кусками площини)

На прикладі 3 – 5вершин

$\delta\overline{Z_0}$ (зовнішньої грані) розміщено вершини 1,2,3,4 графа$K_{2,3}$, а вершини 5 на $\delta\overline{Z_1}$. Тоді

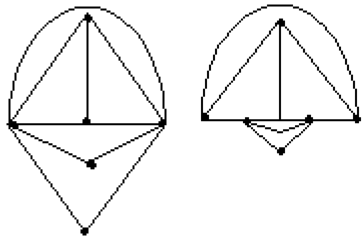


число досяжностей рівня двом для графа $К_{2,3}$, та позначимо $t_{К_{2,3}}(К_{2,3},^0) = 2$

Приклад 4:

Склеїмо ці графи $G_1 i G_2$ по ланцюгам довжини 2 з метою отримання графа з числом досяжності 3 та мінімальним за кількостю ребер (безлімітних ребер).

$t_{G_1}(G^0) = ^{t_{G_2}}(G_2) = 2$



G_3 – це $G_{2+}G_1$ склеїмо по ланцюгам довжини 2(вони ототожнені) $t_{G_3}(G_3^0) = 2,$

В нашому випадку наслідування полягатиме в тому, що синтезований граф має число досяжності множини вершин рівне зменшеній на 1 сумі чисел досяжності множин вершин тих графів, шо синтезувалися

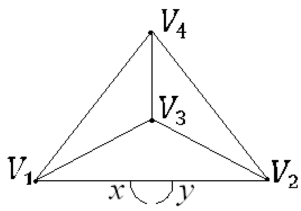
«Лінійний синтез скінченних графів»

Лінійний синтез скінченних графів G_1 та G_2 полягатиме в ототожненні їх по двом простим ланцюгам. L_1 та L_2, де L_i - ланцюга графа G_i

Наприклад $G_1 = K_4, G_2 = K_{2,3}$простих

Множини всіх рвзних ланцюгів для K_4 по довжині розібємо на групи

1) $V_1,(V_1,V_2),V_2 = V_1V_2$
2) $V_1,(V_1V_2),V_2(V_2V_3),V_3 = V_1V_2V_3$

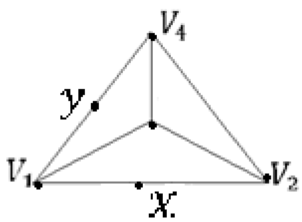


x,y – додаткові вершини

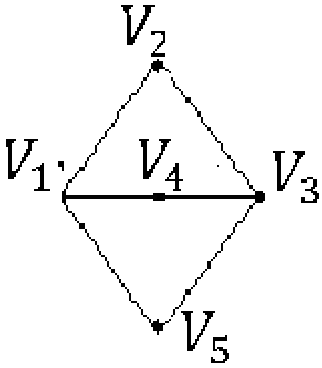
x,y- внутрішні точки, що стали додатковими та вершинами породили ланцюг (xV_2y)

3 Простих ланцюгів довжини три немає, але є під ланцюгові довжини три, а саме

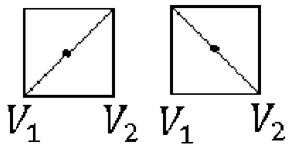
$$X,(V_1V_2),V_2(V_2V_4),V_4(V_4y),y$$



«лінійний синтез»



Множина простих ланцюгів графа Сробиваеться на такі градуси по довжині

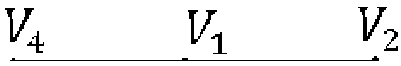


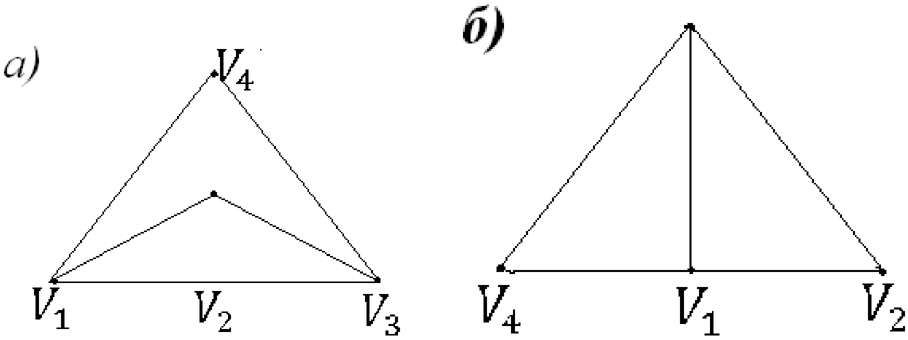
Будемо вважати представниками ребер аж два ребра V_1V_2 та $.V_2V_1$

Ці два ланцюга вважаемо різними з точки зору формального перетину вершини починаючи з пробної лівої вершиниV_1,яка починаючи **?????** в варіанті а здійснюється двома, а в варіанті б- з одніею

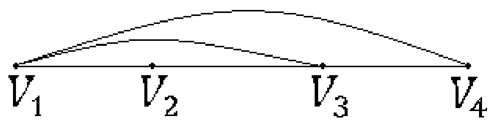
а) $V_1,(C),V_2(V_2V_3),V_3 = V_1V_2V_3$

б) $V_4,(V_4V_1),V_1(V_1V_2),V_2 = V_4V_1V_2$

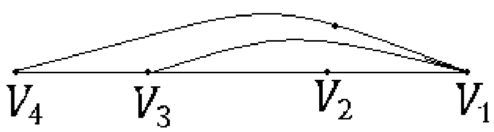




Два представника простих ланцюгів довжини



$\leftarrow V_1V_2V_3V_4$



$\leftarrow V_4V_3V_2V_1$

Є представники простих ланцюгів 3

4 Четверта група складатиметься з двох різних з підланцюгів довжини ;.(немає у $K_{2,3}$ простих ланцюгів), що утворені наступним чином

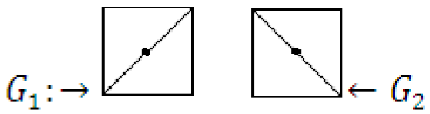
а) х- внутрішня точка ребра V_1V_2

у- внутрішня точка ребра V_4V_1

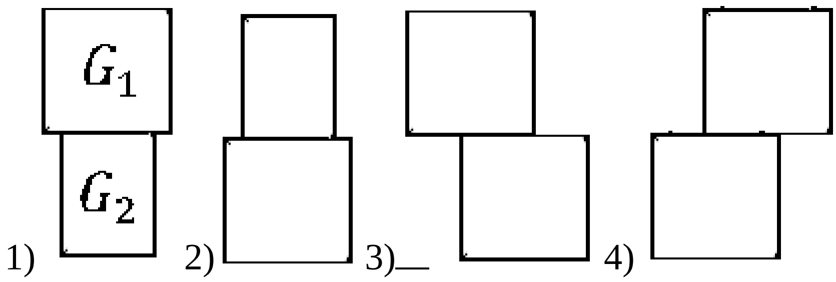
в) х- внутрішня точка ребра V_4V_3

у- внутрішня точка ребра V_1V_4

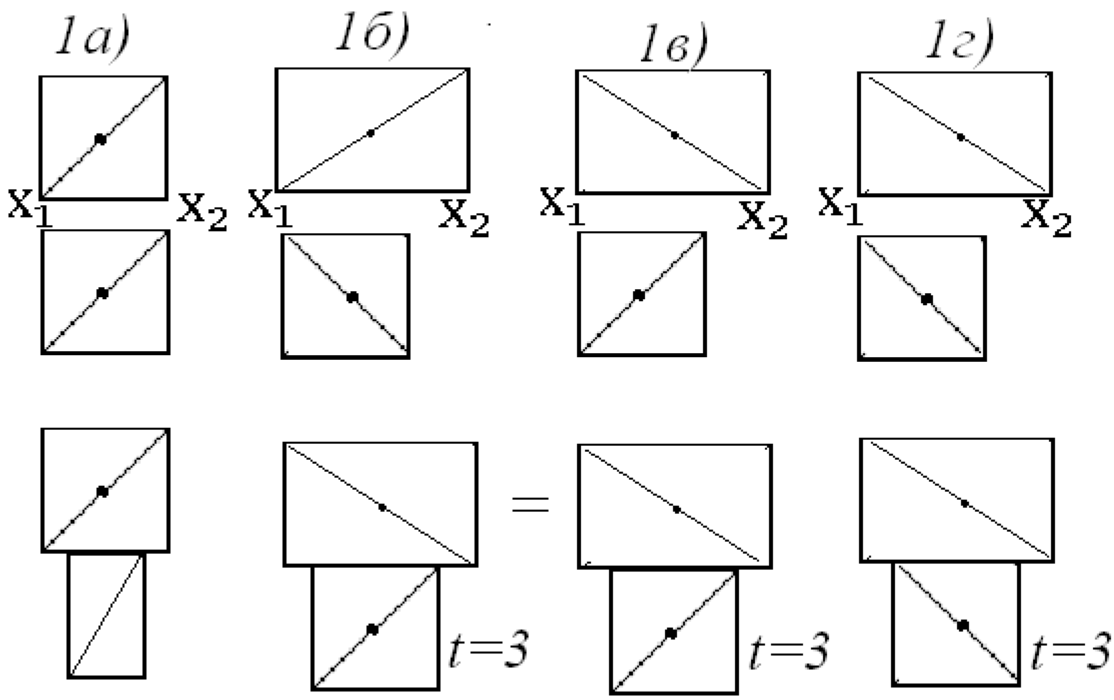
Висновок : лінійний синтез двух графів виконуємо шляхом ототожнення вершин всіх різних пар ланцюгів та під ланцюгів взятих з множини всіх різних ланцюгів графи G_1 та . G_2 (G_1 із списка мінімальних графів , G_2 або $K_{2,3}$або K_4)



Синтезуємо полс дов, 1наступним шляхом



Варіанти:



Виконуємо ототожнення пар вершин$(x_1, 1)$та $(x_2, 2)$ та викинемо парне ребро

T=3 – число досяжності

Результат лінійного варіанта 1(рівне трьом)

«Про лінійне синтезування графів»

Завдання: Побудувати шляхом лінійного синтезу графівG_1 та G_2 , де G_2 це або $K_{2,3}$ або K_4 використовуючи всі різні прості ланцюги для ототожнення де графа G_1 належить до списка мінімальних (в файлі Pilot

Під 3-три мінімальним графом будемо розуміти такий площинний граф у якого немає лінійних ребер при операцій як видалення ребра так і його стискання в точку відносно числа досяжностей множини вершин рівною трьом

Якщо виконувати операцію видалення або стискання будь – якого ребра в точку, графа G, то отримаємо нові графи числам досяжностей $z(> 3)$ (третьому властивостей графів)

Для визначення числа досяжностей треба постановити всі різні рисунки графів на площині.

Різні відносно кількості та якості граней.

Мал.. 2-е вкладення 6е підходять для обчислення

Завдання: виконати синтезування 3- мінімального графів G_1

Із графом $K_{2,3}$

Розв’язок: побудуемо спочатку множини всіх різних простих ланцюжків графів G_1 та G_2

В $K_{2,3}$ такі представники множини простих ланцюгів

1) (Два ребра), два ланцюги довжини 1
2) Два різних ланцюгів довжини
3) Простих ланцюгів довжини 3 е під ланцюг довжини

Задача виконати лінійний синтез графів G_1 та G_2, де

Та $G_2 \in \{К_4, К_{2,3}\}$шляхом ототожнення по всім різним простим ланцюгам.

Розв’язання . Побудуємо множини всіх різних простих ланцюгів графа G_1, беручи до уваги їхне наступне ототожнення з G_2

1)Прості ланцюги довжини 1:

Це всі S різні ребра, до приклеїмо $К_{2,3}$

2)Прості ланцюгом довжини 2:

Однакові ланцюги(3,0,2)=(3,0,1)

Неможливий рисунок ребра (1,6)

3)прості ланцюги довжини 3

г)простий ланцюг довжини 3 проходить через 3,0,5,6 непридатний , що для

Ланцюг 3,0,7,4 має довжину 3 та є непростим, але містить під ланцюг довжини$(x_1, 0, 7, x_2)$,$x_1 \in (0,3), x_2 \in (7,4)$4) прості ланцюги довжини 4.

Властивості планарн**х графів із заданим числом досяжності**

Позначення 1.1 Надалі через G будемо позначати планарний граф, через δ площину. Через $G_0(X)$ або через $(G)_0(X)$ будемо позначати граф, отриманий із графа G шляхом I-підрозділу ребер крапками графа G, що належать цим ребрам і безлічі X.

Пропозиція 1.0.0. Нехай G - не зовнішньопланарний граф, X - безліч крапок графа G. Мають місце наступні твердження :

(1) а) Граф G не зовні планарный тоді тільки тоді, коли існує підграф H графа G гомеоморфный або K_4, або $K_{2,3}$.

б) Існує таке вкладення f, $f: G \to \sigma$ реалізуюче t, де $t = t_G(X)$, $S_G(X) = \{s_i\}_{i=1}^{t}$, що $s \notin S_G(X)$, де s - зовнішня грань графа $f(G)$;

(2) Якщо G - блок, $t_G(G^0) = t$, $S_G(G^0) = \{s_i\}_{i=1}^{t}$, те для кожної пари $(s_i s_j)$, де $i \neq j$, існує найменша по включенню частина H_{ij} графа, що G задовольняє співвідношенню:

(*) $[((G^0 \cap ds_i \subset H_{ij}^0\ ) \wedge (H_{ij}^0 \cap (ds_j - ds_i) \neq 0\ )) \vee$

$[((G^0 \cap ds_j \subset H_{ij}^0\ ) \wedge (H_{ij}^0 \cap (ds_i - ds_j) \neq 0\ )) \wedge$

$(H_{ij} \cong K_4) \vee (H_{ij} \cong K_{2,3})]$

Доказ. Нехай G - не зовнішньо планарный граф, X - безліч крапок графа G. Доведемо твердження (1).

Частина а) твердження 1 випливає з теореми 11.10 [4]. Доведемо частину б). Нехай задане вкладення $f, f: G \to \sigma$, що реалізує t, де $t = t_G(X)$. Безпосередньо з визначення 1.1 випливає співвідношення:

$$(\exists s)(s \notin \sigma(G, f))(s \notin S_G(X)).$$

Припустимо, що α - операція стереографиченого проектування з центром у клітці s. Побудуємо вкладення $f_1 : G \to \sigma$, де $f_1 = \alpha f$. Очевидно, що вкладення f_1 реалізує t, причому клітка s є зовнішньою гранню графа $f_1(G)$, де $\alpha(s_i) = s_i, i = 1(1)t$. Отже, вкладення f_1 задовольняє частини б) твердження(1). Твердження (1) доведено.

Доведемо твердження(2) методом індукції по t.

База індукції: t =2.

Якщо t =2, то в силу теореми 11.10 [4] існує частина H графа, що G задовольняє співвідношенню (*). База доведена. Покладемо, що для $t = n$ твердження (2) доведено.

Обґрунтуємо індукційний крок: $n = n+1$.

Припустимо що в блоці G, де $t_G(G^0) = n+1$ при кожнім $S_G(G^0)$ мається така пари $(s_i, s_j), i \neq j$, для якої не існує частина H графа G задовольняючому співвідношенню (*). Покладемо, що $i = n+1, j = n$. Оскільки G - блок, то існує наименшая по включенню 2-зв'язна частина H графа G, що задовольняє співвідношенню:

$$[((ds_n \subset H^0) \wedge (ds_{n+1} \setminus ds_n) \cap H^0 \neq 0)) \vee ((ds_{n+1} \subset H^0) \wedge (ds_n \setminus ds_{n+1}) \cap H^0 \neq 0))].$$

Розглянемо вкладення $f \mid H, \quad f \mid H : H \to \sigma$. Нехай s'' - зовнішня грань графа $f \mid H(H)$, а через H' - позначимо ту частину підграфа H графа G, що не містить ребер з безлічі $G^1 \bigcap (ds^{''} \bigcap \bigcup_{k=n}^{n+1} ds_k)$. У силу зробленого вище пропозиції маємо $(H^{'})^0 \subset ds^{''}$.

Тому можливо побудувати вкладення $f^{'}$, $f^{'}: G \to \sigma$, у такий спосіб:

а) $f^{'} | G^1 \setminus (H^{'})^1 = f \ | G^1 \setminus (H^{'})^1$;

б) $f^{'}(H^{'} \setminus ((H^{'})^1 \bigcap ds^{''})) \subset s_0$;

де s_0 - зовнішня грань графа $f(G)$.

У результаті одержимо наступне співвідношення:

$f^{'}(G^0) \subset ds^{''} \bigcup (\bigcup_{k=1}^{n-1} ds_k)$, де

$ds^{''} = \bigcup_{k=n}^{n+1} ds_k \setminus \bigcup_{k=n}^{n+1} ((H^{'})^1 \bigcap ds_k$,

$S_G(G^0) = \{s_k\}_{k=1}^{n-1} \bigcup \{s^{''}\}$,

з якого випливає, що $t_G(G^0) \le n$. Одержимо протиріччя умові, що $t = n+1$. Отже, для будь-якої пари (s_i, s_j), $i \ne j$, мається найменша по включенню частина H_{ij} графа, що G задовольняє співвідношенню (*). Доказ пропозиції 1.0.0. закінчено.

Наслідок 1.0.0. Мають місце наступні твердження:

1) Якщо G -1-зв'язний граф, то для кожного 2-компонента G_i графа G, $G_i \subset \varphi^{-1}(G)$, має місце твердження (2) пропозиції 1.0.0.

2) Якщо G - незв'язний граф, то для кожного його 1-компонента виконується твердження 1) дійсного наслідку.

Доказ цих тверджень очевидно.

Позначення 1.3. Будемо позначати через $M(G)$ безліч усіх різних підграфов H графа G, описаних у твердженні (2) пропозиції 1.0.0., а через $M^{'}(G)$- найменше по включенню підмножина безлічі $M(G)$, що складає з найменших по включенню підграфов H_{ij} графа G або частин цих підграфов, що задовольняють наступним умовам:

а) $G^0 \subseteq \bigcup_{\forall H^{'} \in M^{'}(G)} ,(H^{'})^0$;

б) Якщо підграф H_{ij} або його частина гомеоморфны графові K_4, те або усі ребра графа K_4 1-підрозділені, або не одне ребро графа K_4 не 1-підрозділено.

Зауваження.

Надалі якщо не зроблені застереження, будемо вважати що, у відношенні елементів безлічі $M^{'}$ термін «підграф» графа G, не виключає того, що цей елемент може бути частиною графа G.

Приклад.

Розглянемо наступне вкладення графа G в площину:

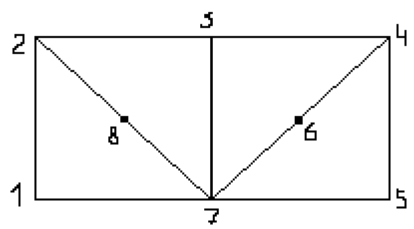


Рис 1.

І виділимо 2 безлічі $S_i = \{s_{ij}\}_{j=1}^{3}, i = 1,2$,

де а) $ds_{11} = \{1,2,7,8\}$, $ds_{12} = \{2,3,7,8\}$, $ds_{13} = \{4,5,6,7\}$;

б) $ds_{21} = \{1,2,7,8\}$, $ds_{22} = \{1,2,3,4,5,6,7\}$, $ds_{23} = ds_{13}$.

Для кожного з них побудуємо безліч $M_i, M_i = M(G)$:

$H_{12}^0 = \{1,2,3,7,8\}$, $H_{21} = \{1,2,7,8\}$, $H_{23}^0 = H_{32}^0$;

$(H_{13}^0)^{'} = \{1,2,6,7,8\}$, $(H_{13}^0)^{''} = \{1,2,4,7,8\}$, $(H_{13}^0)^{'''} = \{1,2,5,7,8\}$,

$H_{13}^0 \in \{(H_{13}^0)^{'}, (H_{13}^0)^{''}, (H_{13}^0)^{'''}\}$,

$H_{32}^0 = \{3,4,5,6,7\}$, $H_{31}^0 = \{1,4,5,6,7\}$,

$M(G) = \{H_{12}, H_{13}^{'}, H_{13}^{''}, H_{13}^{'''}, H_{21}, H_{31}, H_{32}\}$;

$H_{12}^0 = \{1,2,7,8,3\}$, $H_{13}^0 = \{1,2,8,7,6,4,3\}$, $H_{21}^0 = H_{12}^0$,

$H_{23}^0 = \{1,2,3,4,5,7,6\}$, $H_{32}^0 = \{4,5,6,7,3\}$,

$M(G) = \{H_{12}, H_{13}, H_{23}, H_{32}\}$.

Не важко переконається в тім, що приведені вище безлічі а) і б) вичерпують усі неізоморфні безлічі $S_G(G^0)$. У результаті $M^{'}(G) = \{H_{12}, H_{32}\}$.

Наслідок 1.0.1.

Нехай G - блок, $G = (G^0, G^1)$, $t = t_G(G^0)$, кожне ребро якого істотно відносно t при операції видалення, $|M^{'}(G)| = m$.

Тоді мають місце співвідношення:

а) $(\forall u)(\exists H)(u \in G^1)(H \in M'(G))[u \in H^1]$;

б) $(\forall ij)(i \neq j, i, j = 1(1)m)[H_i \cap Hj \neq 0 \Rightarrow p_1(H_i \cap H_j) \leq 1]$.

Доказ. Нехай виконана умова наслідку 1.0.1.

Доведемо співвідношення 1). Розглянемо вкладення $f^{'}$ $f^{'}: G \to \sigma$, що реалізує t, $t = t_G(G^0)$. Нехай u - довільне ребро графа G. Будь-яка безліч $S_G(G^0)$, $S_G(G^0) = \{s_i\}_{i=1}^t$ задовольняє одному з наступних співвідношень :

а) $f(u) \subset ds_1$,

б) $f(u) \subseteq \bigcap_{i=1}^{2} ds_i$.

Згідно твердження(2) пропозиції 1.0.0. для будь-якої пари $(si, sj),\ j = 2(1)t$, мається підграф H_{ij} гомеоморфний або K_4, або K_{23} і задовольняючому співвідношенню:

$$(G^0 \bigcap ds_1 \subset H_{ij}) \wedge (H_{ij} \bigcap (ds_j \setminus \bigcup_{\substack{k=o \\ k=1}}^{t} ds_k))$$

Таким чином, для кожного u, $u \in G'$, мається такий підграф $H_{ij}, H_{ij} \in M(G)$, що $u \in H_{ij}'$. Тоді існує підмножина $M'(G)$безлічі $M(G)$, що містить найменше число таких підграфов H_{ij}.

Доказ твердження а) закінчено.

Доведемо твердження б).

З визначення безлічі $M'(G)$випливає, що будь-які його два різних елементи $H_i, H_j, i \neq j$, що має не порожнє перетинання, можуть мати не більш одного загального циклу.

Дійсно, якщо запропонувати, що елементи H_i, H_j безлічі $M'(G)$мають два загальних цикли, то в силу умов а), б) приведених у позначенні 1.3, те ці елементи гомеоморфны графові K_4, у якого жодне ребро не 1-підрозділене. Тоді можливо побудувати безліч M , де $M = M'(G) \setminus \{H_i\}$, що задовольняє умовам а), б) позначення 1.3., т.е одержимо протиріччя умові для безлічі $M'(G)$. Припущення не вірне.

Твердження б) доведено.

Доказ наслідку закінчений.

<u>Наслідок 1.0.2.</u> Нехай G - блок, $M'(G) = \{H_i\}_{i=1}^{m}$.

Мають місце наступні твердження :

а) Елементи безлічі $M^{'}(G)$ можуть мати не більш одного простого циклу у своєму перетинанні;

б) Якщо $t_G(G^0)=3, m=2$, те виконується одне з наступних співвідношень:

б1) $\bigcap_{t=1}^{2} H_1 = \sum_{j=1}^{n} C_G^{nj}(a_j, b_j)$, де $n \geq 0$;

б1) існує частина графа G гомеоморфна графові K_4.

<u>Доказ.</u> Нехай G - блок, $M^{'}(G)=\{H_i\}_{i=1}^{m}$ $m>1$. Неважко побачити незалежність співвідношення 2) наслідки 1.0.1. від умови істотності кожного ребра графа щодо числа при операції видалення ребра.

Твердження а) доведено.

Доведемо співвідношення б). Покладемо, що $m>2$ $t_G(G^0)=3,$

У силу твердження а) можливі тільки наступні випадки:

а) $p_1(\bigcap_{i=1}^{2} H_i)=0$,

б) $p_1(\bigcap_{i=1}^{2} H_i) \neq 0$

Розглянемо випадок а), якщо графи H_i, $i=1,2$ не мають загальних простих циклів, то виконується співвідношення б1), де $C_G^{nj}(a_j, b_j)$ - простий ланцюг графа G довжини $n_j, n_j \geq 0$.

Що було потрібно довести.

Розглянемо випадок б). Відповідно до визначення безлічі $M^{'}(G)$ можливі тільки 3 різні пари, що складають которых будуть частини H_i графа G - гомеоморфные графові $K_{2,3}$ або ізоморфні графові K_4. Неважко переконається в тім, що наявність пари (K_4, K_4) суперечить умові, т.до об'єднання двох графів K_4 із загальним простим циклом містить частина гомеоморфную графові $K_{2,3}$, тобто одержимо включення

$\bigcup_{i=1}^{2} H_i^0 \subset H^0$, що суперечить умові.

Інші пари (H, H), (K_4, H) можуть мати місце тільки тоді, коли

$\bigcup_{i=1}^{2} H_i$, містить частина гомеоморфную графові K'_4.

Дійсно, якщо простий цикл $Z, Z \in \bigcap_{i=1}^{2} H_i$ має довжину n, $n \geq 8$, те ля пари (H, H), одержимо що граф $\bigcup_{i=1}^{2} H_i$, де $H_i \cong H$, гомеоморфный графові K'_4, а для іншої пари (K_4, H), одержимо що граф

$\bigcup_{i=1}^{2} H_i$ має частина гомеоморфную графові K_4. Тим самим доведене співвідношення б2). Твердження б) доведено. Доказ наслідку 1.0.2 закінчено.

<u>Наслідок 1.0.3.</u> Нехай G -блок, $M'(G) = \{H_i\}_{i=1}^{3}$.

Якщо елементи безлічі $M'(G)$ мають загальний простий цикл Z, то мають місце наступні співвідношення:

а) $p_1(\bigcap_{i=1}^{2} H_i) = 0$;

б) якщо $Z \subset H_i \bigcap H_j$, те графи $H_k \bigcap H_j, H_k \bigcap H_j$ не містять простих циклів і $H_i \cong H_j \cong K_{2,3}$, $\{i, j, k\} = \{1,2,3\}$;

в) якщо $H \cong K_{2,3}$, те $Z \subset H_i \bigcap (H_j \bigcup H_k)$, де Z цикл графа G, що є границею зовнішньої грані для графа $f \mid H_j \bigcup H_k (H_j \bigcup H_k)$, де вкладення $f, f : G \to \sigma$, реалізує $t_G(G^0)$.

Доказ. Нехай G-блок, $t_G(G^0)=3$,

$M^{'}=\{H_i\}_{i-1}^{3}$,

Що означає, що граф G не містить частини гомеоморфной графові $K_4^{'}$. Покладемо, що для деякого графа G при кожнім $M^{'}(G)$ маються елементи, наприклад $H_i, i=1,2$, що володіють загальним простим циклом Z. Відповідно до визначення безлічі $M^{'}(G)$, елементи якого або гомеоморфні графові $K_{2,3}$, (у цьому випадку будемо позначати них через H), або ізоморфні графові K_4. Представляються можливими тільки наступні випадки.

1) Існує Z, $Z \subset \bigcap_{i=1}^{3} H_i$;

2) тільки одна пара елементів безлічі $M^{'}(G)$ має загальний простий цикл;

3) у точності дві пари елементів мають загальні цикли приналежні:

3а) одному елементові;

3б) різним елементам;

4)три пари мають різні загальні цикли.

Розглянемо випадок 1). Маємо наступні 4 різних варіанти для трійок елементів безлічі $M^{'}(G)$, що вичерпують усі можливості:

10) (H,H,H) 12) (H,K_4,K_4)

11) (H,H,K_4) 13) (K_4,K_4,K_4).

Неважко переконається в тім, що в кожнім з цих варіантів можна знайти такі 2 частини $H_i, i=1,2$ графа G, що гомеоморфные або $K_{2,3}$, або ізоморфні графові K_4 і мають властивість: $G^0 \subseteq \bigcup_{i=1}^{2} H_i$.

Тим самим одержимо протиріччя умові, що $|M^{'}(G)|=3$.

Випадок 1) неможливий.

Розглянемо три різні пари елементів безлічі $M^{'}(G)$.

Нехай має місце випадок 2). Для пари, наприклад (H_1, H_2), що має загальної простий цикл можливі тільки наступні варіанти:

20) (H, H); 21) (H, K_4); 22) (K_4, K_4).

Відповідно до визначення безлічі $M^{'}(G)$варіанти 21) і 22) є неможливими, т.до у кожнім випадку мається частина $(H^{'})$ графа G обладающая властивістю:

$$(H^{'} \cong K_{2,3}) \wedge (\bigcup_{i=1}^{2} H_i \subseteq H^{'}),$$

що суперечить умові:

$$\left| M^{'}(G) \right| = 3.$$

Представляється можливим варіант 20) , що дає співвідношення б) дійсного наслідку.

Випадок 2) розглянутий.

Нехай має місце випадок 3а). Покладемо, що дві різні пари елементів $H_i, H_i \in M^{'}(G),$ мають загальні прості ?:? для своїх складових, приналежному одному елементові.

Дійсно, якщо припустити, що існують дві пари, наприклад: $(H_1, H_2), (H_1, H_3)$, що володіють наступною властивістю:

$$(\forall_i)(i = 1{,}2)(z_i \subset H_1 \cap H_i)$$

z_i - простий цикл елемента H_1 те неважко переконати, що це неможливе припущення.

Розглянемо наступні можливі під випадки для випадку 3а) :

3а1) $H_1 \cong K_{2,3}$;   3а2) $H_1 \approx K_4$.

У кожнім подслучае має місце співвідношення $G^0 \subseteq \bigcup_{i=1}^{2} H_i^0$, що суперечить умові. Обоє подслучая неможливі. Такому образом випадок 3а) неможливий.

Нехай має випадок 3б). Покладемо, що дві різні пари елементів $H_i, H_i \in M^{'}(G),$ мають загальні цикли для своїх складових, приналежним різним елементам безлічі $M^{'}(G)$.

Припустимо, що дві пари, наприклад: $(H_1, H_2), (H_2, H_3)$, мають властивість:

$$(\forall_i)(i = 1{,}2)[(z_1 \subset \bigcap_{i=1}^{2} H_i) \wedge (z_2 \subset \bigcap_{j=2}^{3} H_j)],$$

$$z_1 \subset H_3, z_2 \not\subset H_1.$$

Використовуючи ті ж доводи, що і для випадку 3а) ми одержимо, що випадок 3б) неможливий.

Нехай має місце випадок 4). Оскільки серед трьох пар, що володіють попарно загальними циклами, маються 2 пари, що володіють загальними циклами, що належать або одному елементові, або різним елементам, те випадок 4) зводиться до випадку 3).

Випадок 4) неможливий.

З приведених вище міркувань випливає наступне: якщо $H_i \cong K_{2,3}$, те $z \subset H_i \cap (H_j \cup H_k)$, де Z- цикл графа G, що є границею зовнішньої грані для графа $f \mid H_j \cup H_k (H_j \cup H_k)$, де вкладення f , $f : G \to \sigma$, реалізує $t_G(G^0)$, $\{i, j, k\} = \{1{,}2{,}3\}$.

Доказ следсьвия закінчений.

Наслідок 1.0.4. Нехай $K_4^{'} = (K_4)_0(K_4^0)$

1) Граф G - задовольняє співвідношенню:
$(\forall u)(u \in G^1)[t_{G\backslash u}(G^0) = t_G(G^0) - 1 = 1]$

тоді і тільки тоді, коли граф G гомеоморфный або $K_{2,3}$, або K_4, причому $G \neq K_4^{'}$.

2) Якщо граф G гомеоморфный $K_4^{'}$, те має місце:

$(\forall u)(u \in G^1)[t_{G\backslash u}(G^0) = t_G(G^0) - 1 = 2]$.

Доказ.

Доведемо1). Нехай граф G задовольняє зазначеному співвідношенню, тобто кожне ребро графа G істотно відносно $t_G = (G^0)$ при операції видалення. У силу пропозиції 1.0.0. мається частина H графа G, гомеоморфная або K_4, або $K_{2,3}$.

Припустимо, що має місце співвідношення: $G^1 \setminus H^1 \neq 0$. Розглянемо ребро u, $u \in G^1 \setminus H^1$ і граф G^1, $G_1 = G \setminus u$. Тому що $H \subset G_1$, $t_G(G^0) = 2$, те в силу твердження 2 пропозиції 1.0.0. маємо рівність: $t_{G_1}(G_1^0) = 2$, що суперечить приведеному співвідношенню. Припущення невірне. Отже, граф G гомеоморфний або $K_{2,3}$, або K_4. Нехай граф G гомеоморфний K_4 або $K_{2,3}$, причому граф $G \neq K_4^{'}$. Очевидно, що $t_G(G^0) = 2$.

Твердження 1) доведено.

Доведемо 2). Нехай $G \cong K_4^{'}$, тобто граф G може бути отриманий із графа K_4, шляхом 1-підрозділу всіх його ребер. Тому що існує єдине неізоморфне мінімальне вкладення графа G в σ, те має місце: $t_G(G^0) = 3$.

Неважко переконається в тім, що виконується співвідношення:

$(\forall u)(u \in G^1)(t_{G\backslash u}(G^0) = 2)$.

Доказ наслідку закінчений.

Пропозиція 1.0.1. Нехай G - плоский граф, $t_G(G^0) = t$, $t > 1$.

Для будь-якого ребра u, $u \in G^1$, виконується нерівність:

$t - 1 \le t_{G\backslash u}(G^0) \le t$.

Доказ. Нехай G-плоский граф, $t_G(G^0) = t$, $t > 1$, $u \in G^1$, $H = G \setminus u$. Оскільки $H^0 \subseteq G^0$, та нерівність $t_H(H^0) \ge t - 1$ справедлива для кожного ребра u графа G. Використовуємо метод доказу від противного.

Припущення 0. Нехай для деякого u, $u = (a_1 a_2)$, виконується нерівність: $t_H(H^0) < t - 1$, де $t_H(H^0) = t^{'}$

Доведемо лему 01:

Кожне вкладення $f^{'}$, $f^{'} : H \to \sigma$, що реалізує $t^{'}$ задовольняє співвідношенню:

$(\forall i, s)(s_i \in S_h(H^0))(s \in \sigma(H, f^{'})(i = 1,2)[(f^{'}(a_i) \in ds_i) \wedge (f^{'}(du) \not\subset ds)]$.

Припустимо, що мається таке вкладення f, $f : H \to \sigma$, що реалізує $t^{'}$, що задовольняє співвідношенню:

$f(du) \subset ds$, де $s \in \sigma(H, f)$. Побудуємо вкладення f_1, $f_1 : G \to \sigma$, так, щоб виконувалися наступні умови:

1) $f_1 | H = f | H$,

2) $f_1(u) \subset s$.

Одержимо наступне співвідношення $f_1(G^0) \subset \bigcup_{i=1}^{t'} ds_i$, що суперечить умові нашої пропозиції. Припущення невірне. Отже при будь-якому вкладенні f', $f': H \to \sigma$, що реалізує t', має місце співвідношення: $f'(a_i) \in ds_i$, де $s_i \in S_H(H^0)$, $i = 1,2$. Лема доведена.

Відповідно до припущення можливі тільки наступні випадки:

А) $p_G(a_1) \geq p_G(a_2) > 2$;

Б) $(p_G(a_1) \geq p_G(a_2)) \wedge (p_G(a_2) = 2)$.

Розглянемо випадок А). Покладемо, що вкладення $f': H \to \sigma$, реалізує t' $S_H(H^0) = \{s_i\}_{i=1}^{t'}$. Згідно леми 01 маємо, що $f'(a_i) \in ds_i$, $i = 1,2$.

Нехай M - найменший по включенню блок графа H, що містить $\bigcup_{i=1}^{2} ds_i$. Підграф M графа H містить простий ланцюг C_2, $C_2 = C_G(b_1, b_2)$, таку, що пари вершин $(b_1, b_2), (a_1, a_2)$ розділяють один одного на площині і належать простому циклові Z підграфа M.

Доведемо лему 02:

Не існує простого ланцюга C_3, $C_3 = C_G(C_1, C_2)$, такої, що пари вершин $(a_1, a_2), (b_1, b_2), (c_1, c_2)$ належать циклові Z і розділяють один одного на площині.

Припустимо, що мається простий ланцюг C_3, описана в умові. Розглянемо підграф N, $N = Z \cup C_2 \cup C_3$ графа H. Оскільки $N \cong K_4$ і вершини a_1, a_2 є крапками несуміжних ребер графа N, те маємо рівність: $t_H(\{a_i\}_{i=1}^{2}) = 2$. Розглянемо наступне φ - перетворення

$$\varphi(H+St_2(y_0),\sum_{i=1}^{2}(a_i+y_i))=(Y,\{a_i^*\}_{i=1}^2)\text{ ,де }St_2^0(y_0)=\{y_i\}_{i=0}^2 .$$

Алгоритм обчислення числа досяжності множини точок графа

Розглянемо задачу знаходження числа досяжності $t_G(X)$деякої спеціальної множинаі X - точок плоского 2-зв'язного графа Gтакого, що при будь-якому вкладенні $f:G\to\sigma$ ніякі два елементи множинаі X не належать границі якої-небудь 2-клітки. Згідно визначення цього числа один зі шляхів рішення цієї задачі зводиться до побудови множинаі всіх неізоморфних вкладень графа G в. σ

На підставі наслідку 2.0 теореми 2.2 можна побудувати наступний алгоритм перебування числа $t_G(X)$;

0. У графі Gвиділимо множина М, що складається з усіх його різних підграфів H, що відносно X_H стягаються до одному з графів, описаних у визначенні 2.3.

1. Вибираємо підмножини M_i множинаі М,| M_i|=m,

$$M_i=\left\{H_{k\ k+1}^{(i)}\right\}_{k=1}^{mi-1}\cup\left\{H_{1m_i}^{(i)}\right\},$$

елементи яких мають властивість:

для кожної з інших пар $(i',j'),i'<j'$ існує підграф Hi'j' графа G, де

$H_{i'j'}\in$M\$M_i$, такий, що $H_{i'j'}\subseteq\left(\bigcup_{k=1}^{m_i-1}H_{k\ k=1}^{(i)}\right)\cup H_{1m_i}^{(i)}$, причому $H_{i'j'}$ задовольняє визначенню 2.3.

Будемо називати підмножину M_i покриттям довжини m_i, множинаі X , причому назвемо покриття M_i , M_j різними, якщо $|M_i|\neq|M_j|$.

2. Побудуємо послідовність M .що складається з усіх різних покрить множинаах. Тому що G- - кінцевий граф і розглядаються різні підграф H , те M скінчена послідовність. Тому можливо упорядкувати елементи послідовності M по убуванню їхньої довжини, а також перенумерувати них.

3. Якщо отримана послідовність M має довжину P, то M_p - покриття мінімальної довжини. Тоді виконується рівність: $t_G(X)=|M_p|$. Опис алгоритму закінчений.

На підстав наслідків 2.1 і 2.2 можливо побудувати аналогічні алгоритми рішення задачі обчислення числа досяжності деякої множина X , що складає з точок однозв'язного графа або незв'язного графа.

Для ілюстрації нашого алгоритму звернемося до прикладів 2.1,2.2 і знайдемо числа $t_{G_i}(G_i \cap X), i=1(1)4,$ для кожного приклада. Розглянемо граф G_1 із приклада 2.1.

Нехай $X=\{x_i\}_{i=1}^{4}$. і задана множина S , $S=\{s_i\}_{i=1}^{4}, X_i = X \cap \partial s_i, i=1(1)4$. Для кожної пари (s_i, s_j) , де $i<j, i, j=1(1)4$ побудуємо H_{ij} - частковий підграф графа G_1 , описаний у визначенні 2.3, що стягається відносно X_{Hij} до графа $K_4^{(1)}$ Неважко переконатися в тім, що при будь-якому вкладенні $f: G_1 \to \sigma$ для будь-яких $x_i, x_j,$ де $x_i \neq x_j$, $f(\{x_i, x_j\}) \not\subset \partial s, s \in \sigma(G_1, f)$. Дійсно, для будь-якої пари $(x_i, x_j), i \neq j,$ існує підграф H_{ij}, , такий, що $H_{ij} \cong K_4^{(1)}$. Маємо M_1 покриття множина X , де $M_1=\{H_{12}^{(1)}, H_{14}^{(1)}, H_{23}^{(1)}, H_{34}^{(1)}\}$

причому для всіх інших пар $(i, j), i<j, i, j=1(1)4.$, існує такий підграф H_{ij} графа G_1 що $H_{ij} = \bigcup_{\forall H \in m_1} H$

де H_{ij} описаний у визначенні 2.3. Очевидно, що M_1 містить найменше число елементів. Згідно наслідку 2.0 маємо рівність: $t_{G_1}(X \cap G_1) = |M_1| = 4$.

Аналогічно можна довести, що

Алгоритм встановлення 3-властивості площинних графів

Визначення 3.1. Будемо говорити, що граф G має 3-властивість, якщо має місце нерівність: (t G^o)≥ 3.

Використовуючи визначення 3-мінімального плоского графа Сг , а також список таких неізоморфних графів, приведений у додатку дійсної роботи, одержимо наступну лему 3.1;

Якщо граф G- має 3-властивість, то існує частина H графа G гомеоморфна деякому 3-мінімальному графові.

<u>Доведення</u>. Нехай граф G- задовольняє нерівності: t(G^o)≥ 3. Перетворимо граф G у граф G_1 шляхом видалення всіх його ребер несуттєвих відносно t(G^o)≥ 3 т.е. таких ребер u, u ∈ G^1 , що задовольняють рівності: $t_{\setminus u}(G^\circ)=t(G^\circ)$.

Можливо, що граф G_1 буде мати ізольовані вершини, що є несуттєвими при визначенні числа $t(G^0{}_1)$. Очевидно, що граф G_1 задовольняє наступному співвідношенню:

$$(\forall u)(u \in G_1^1)(t_{G\backslash u}(G_1^0) \quad 2)$$

З визначення 1.3 випливає, що граф G_1, гомеоморфний 3-мінімальному графові. Для того, щоб одержати з графа G_1 3- мінімальний граф варто виконати операцію

ототожнення кожної вершини a, де $a \in$М, де

$$\text{М}=\left\{\forall a / t_{G_1}\left(G_1 \setminus \{a\}\right)=3\right\}, (\ )$$

або з однієї із суміжних вершин множини М , або з множини $G_1 \backslash$М . Що було потрібно довести. Доведення леми закінчений.

Має місце наступна теорема 3.0.I;

Граф G має 3-властивість тоді і тільки тоді, коли існує частина графа G гомеоморфна 3-мінімальному графові.

Доведення. Якщо граф G має 3-властивість, то в силу леми 3.1 мається частина графа G гомеоморфна деякому 3-мінімальному графові, з числа приведених у додатку 2. Якщо граф G містить частина H гомеоморфну 3-мінімальному графові, то в силу нерівності: $t_G(G^0) \geq t_H(H^0)$ одержимо, що граф G- має 3-властивість.

Що було потрібно довести. Доведення теореми закінчений.

У зв'язку з цим виникає задача про існування алгоритму для встановлення 3-властивості вхідного графа G- за поліноміальний час. Мають місце наступні леми.

Лема 3.2. Якщо існує простий ланцюг $C, C = C_G^m(a,b), m \geq 2, (a,b) \notin G^1$

задовольняючій умові:

$$(\forall v)(v \in C^0 \setminus \{a,b\})\left[(\rho_G(v)=2) \wedge (\rho_G(a) \geq \rho_G(b) > 2) \wedge (w(G) > 1)\right]$$

те мається простий цикл Z , $\{a,b\} \subset z^0$, де $z^0 = \{z_i\}_{i=1}^n$, $G_1 = (G^0, G^1 \setminus C^1), n > 3$ такий, що підграф $G[zUC]$ містить частину гомеоморфну графові $K_{2,3}$.

Лема 3.3. Нехай $a_0 \in G^0, \rho_G(a_0) = n, St_G(a_0)$ - зірка графа G з центром у вершині a_0 , де $St_G^0(a_0) = \{a_i\}_{i=0}^n$, а ребра-промені можуть бути 1-підрозділені вершинами ступеня 2;

$$S_1 = \{a_i\}_{i=1}^n, \rho_G(a_i) > 2, i = 1(1)n$$

Якщо має місце співвідношення: $(\rho_G(a_0) = n) \wedge (\omega(G \setminus St_G^1(a_0) > 1),$ то існує такий простий цикл Z графа G , де $|z^0| = m, S_1 \supset z^0$ що підграф $G[z^0 U St_G^0(a^0)]$, або гомеоморфний графові K_4, або містить частина гомеоморфну графові ДО$_4$. Доведення цих лем 3.2, 3.3 очевидно. Помітимо, що описаний у їхніх умовах підграф $G[z^0]$ - це простий цикл довжини $|z^0|$ з можливими діагоналями.

<u>Позначення 3.2.</u> Через M_i позначимо найбільше по включенню множина $\{z_{ij}\}_{j=1}^{|M_i|}$ усіх різних простих циклів z_{ij} графа G , описаних у лемі 3. $i+1$, де i=1,2, і задовольняючим умовам:

$$1).(\forall j')(j') = 1(1)|M_i|)\left(z_{ij'} \not\subset \bigcup_{\substack{j \neq j' \\ j=1}}^{|M_i|} \right);$$

$$2).\left(\bigcup_{j=1}^{|M_1|} z_{ij}^1 \cap \bigcup_{j=1}^{|M_2|} St_G^1(vj) \right) = \varnothing,$$

де $St_G(v_j)$- зірка графа G з центром у крапці v_j , зв'язана з z_{2j} і описана в лемі 3.3.

Позначимо через O_n φ-образ графа $St(v) + z$ де $z^0 = \{z_i\}_{i=1}^n$, z - простий цикл довжини n , $St_G(v) = \{v\} \cup \{v_i\}_{i=1}^n$,якщо φ -перетворення задане в такий спосіб:

$$\varphi\left(z+St(v),\sum_{i=1}^{n}(z_1+v_i)\right)=\left(O_n,\{z_i\}_{i=1}^{n}\right),$$

і назвемо граф On - колесом з n -спицями.

Лема 3.4. Якщо M_i - множина, описана в позначенні 3.2, таке, що $\sum_{i=1}^{2}|M_i|=k$ те має місце нерівність: $t_G\left(G^0\right)\geq k$.

Доведення. У силу прийнятого вище позначення 3.2 маємо, що з кожним елементом множини, M_i описаного в позначенні 3.2, зв'язана однозначним образом деяка частина графа G або його підграф, гомеоморфна графові K_4 або графові $K_{2,3}$. Якщо $\sum_{i=1}^{2}M_i=k,$ то в G мається до-штук таких частин графа G не мають один з одним

загальних простих циклів і гомеоморфних або графові K_4, або графові $K_{2,3}$. Кожний із графів K_4 , K2,3 має число досяжності множинаі вершин більше I, тому $t_G\left(G^0\right)$

- число досяжності графа G задовольняє нерівності: $t_G\left(G^0\right)\geq k$.
.Доведення леми 3.4 закінчено.

Лема 3.5. . Нехай $H_i\subset G,H_i\cong O$ або $H_i\cong K_{2,3}$, де i = 1,2. , n >2. Якщо має місце співвідношення: $\left(\sum_{i=1}^{2}|M_i|=2\right)\wedge\left(w\left(\bigcup_{i=1}^{2}H_i\right)\succ 0\right)$, то існує не більш одного простого ланцюга З , $C=C_G^k(a,b)$, довжини до , до $\geq$ 2. , де $(a,b)\notin G^1$, $\{a,b\}\subset\bigcup_{i=1}^{2}H_i$, такий, що задовольняє умові: $C^1\bigcap\left(\bigcup_{i=1}^{2}H_i^1\right)=\varnothing$.

Доведення. Нехай $\sum_{i=1}^{2}.|M_i|=2,\{H_i\}_{i=1}^{2}$ множина підграфів графа G або його частин гомеоморфних або $K_{2,3}$ або K_4, кожний з H_i містить точно один простий цикл, що є елементом одного з множествM_i, i=1,2. . Покладемо, що $\bigcup_{i=1}^{2}Hi$ - зв'язний граф. Для Доведенняу використовуємо метод від противного .

Припустимо, що існує множина $\{C_j\}_{j=1}^{m}$, де m>1 , що складається з простих ланцюгів C_j , $C_j = C_G^{mj}\left(a_j, b_j\right)$ де $m_j \geq 2$ для всіх j,j=1(1)m. , $\{a_j, b_j\}_{j=1}^{m} \subset \bigcup_{i=1}^{2} H_i^0$, таких, що задовольняють співвідношенню:

$$\left(G^1 \cap \left\{\left(a_j b_j\right)\right\}_{j=1}^{m} = \varnothing\right) \wedge \left(\bigcup_{j=1}^{m} C_j^1 \cap \bigcup_{i=2}^{2} H_i^1 = \varnothing\right)$$

Тоді існує простий цикл Z графа G , що містить одну з простих ланцюгів C_j, нехай $j=1$, а також простий ланцюг З , $C = C_G\left(a_1, b_1\right)$, таку, що

$$\left(\{a_2 b_2\} \subset C^0\right) \wedge \left(C \subset \bigcup_{i=1}^{2} z_i\right)$$

Неважко бачити, що $G\left[z^0\right] \cup C_2$ - підграф графа G - або його частина гомеоморфна $ДО_{2,3}$. Це означає, що $z \in M_1$.Тим самим одержимо протиріччя умові, що $\sum_{i=1}^{2}\left|M_i\right| = 2$

Припущення невірне. Отже існує не більш одного простого ланцюга С, описаної в умові леми. Доведення закінчений.

<u>Алгоритм ДО4.</u>

0) $H_2 := \varnothing, Z := \varnothing, F := \varnothing;$

1) Вибираємо вершину V , так, щоб існувала зірка $St_G^n(v)$ графа G з центром в v и n ребрами, n>2, (можливо, що деякі з них виявляться 1-підрозділеними), така, що $w\left(G \setminus \left(St_G^n(v)\right)^1 \setminus \{v\}\right) \succ 1$. Якщо такої вершини ні, то переходимо до мітки 4).

2) Знайдемо вершини простого циклу $z, z = z^n$, у такий спосіб:

а) вибираємо $c, c \in M$, де $M = G^0 \setminus \left(St_G(v) \cup Z\right)$;

в) $F := F + (cv)$

якщо $\gamma(F) = 0$, то: заносимо вершину С в список Z ,

$M := M \setminus Z$ переходимо до мітки а);

якщо $\gamma(F) \succ 0$, то: $M := M \setminus \{c\}$

якщо $M = \varnothing$, те переходимо до мітки 3)

якщо $M \neq \varnothing$, те переходимо до мітки а).

з) $z^0 = Z \bigcup S_1$

де S - множина кінцевих вершин зірки $St_G(v)$, відмінних від v і ступені, що має, більше 2. Тут $G\left[z^0\right]$-простий цикл довжини $m, m = |Z| + n$, (можливі діагоналі циклу не розглядаємо).

$$H_2 := G\left[z^0\right] \bigcup St_G^n(v)$$

4) Якщо $H_2 := \varnothing$ те видаємо "НІ"; інакше:

$$H_2 = G\left[z^0\right] \bigcup St_G^{n(v)}$$

кінець роботи.

<u>Алгоритм ДО23</u>

0) $H_2 := \varnothing, Z := \varnothing, F := \varnothing;$

I) Вибираємо ланцюг C , що задовольняє умові леми 3.2, де $C = C_G(a,b)$, $(ab) \notin G^1$. Якщо такий ні, то переходимо до мітки 4).

2) Знайдемо вершини простого циклу $z, z = z^n$, у такий спосіб:

а) вибираємо вершину g , $g \in C^0 \setminus \{a,b\};$

б) вибираємо вершину $c, c \in G^0 \setminus \left(C^0 \bigcup Z\right);$

в) $F := F + (gc)$

якщо $\gamma(F) = 0$, те $c \to Z$ тобто заносимо c в список Z, переходимо до мітки б);

якщо $\gamma(F) \succ 0$, то $G^0 := G^0\left(\{c\} \bigcup C^0 \bigcup Z\right),$

якщо $\left|G^0\right| = 0$, те переходимо до мітки 3);

інакше: переходимо до мітки б).

3) $z^0 = Z \bigcup \{a,b\}$,

де a,b - кінцеві вершини ланцюга $C, G\left[z^0\right]$ простий цикл довжини n, без діагоналей $n = |Z| + 2$, (можливі діагоналі цього циклу відкинемо). Якщо $n \succ 3$, то

$H_1 := G\left[z^0\right] \bigcup C$, видаємо відповідь: " $H_1 =$`$H_1$`"; переходимо до мітки кінець. Інакше:

4) Видаємо відповідь "НІ". Кінець роботи.

Теорема 3.1. Алгоритми ДО4, ДО23 є коректними і мають поліноміальну складність.

Доведення. Неважко бачити, що в підставі алгоритмів ДО4, ДО23 лежить перевірка наявності властивостей 2-связности або площинності вхідного графа F . Властивість 2-связности

вхідного графа F можна знайти за $0\left(\left|F^0\right|\right)$ операцій за допомогою алгоритму, описаного в роботі $[27]$, а властивість планарности можна знайти за $0\left(\left|F^0\right|\right)$ операцій за допомогою алгоритму, описаного в роботі $[28]$, де $\left|F^0\right| = \left|G^0\right|$. Доведення теореми закінчений. а

Алгоритм А. / обчислимо $K, K = \sum_{i=1}^{2} |M_i|$/

о) $M_i := \varnothing, F := G,$

$Z := \varnothing, K := 0$/ для результату /

$i := 0;$

I) $i := i+1, v := v_i$, де $\{v_i\}_{i=1}^{|F|} = F^0 \setminus Z;$

Якщо $|F \setminus Z| = 0,$ то переходимо до мітки 4).

I.a) чи Існує зірка $St_F(v)$ графа F з центром в v і n -ребрами, де $n = \rho_1(v),$, що можливо І-підрозділені, така, що H_2, де $H_2 = St_F(v) \bigcup z,$ містить частина графа G гомеоморфну графові O_n ? (використовуємо алгоритм ДО4).

Якщо "є", то: $v \to Z$,

$z \to M_2$

$F := G \setminus \left(St^0(v) \setminus S \right)$

перехід до мітки 2); де S_1 - множина усіх кінцевих вершин зірки $St_F(v)$;

Інакше:

1.б) за допомогою алгоритму ДО23 перевіримо існування простого ланцюга $C, C = C_G(a,b)$,де $(a,b) \notin G^1$, такий, що $v \in C^0 \setminus \{a,b\}$, де підграф $H, H = C \bigcup z$ - містить частина гомеоморфную $ДО_{2,3}$.

Якщо "є" то: $v \to Z$,

$F := G \setminus \left(C \setminus \{a,b\} \right)$,

$z \to M_1$;

переходимо до мітки 2). Інакше переходимо до мітки кінець).

2) $j := j + 1$;

$uj \in F^0 \setminus Z$; Якщо $\left| F^0 \setminus Z \right| = 0$, то переходимо до мітки 4).

2.а) За допомогою алгоритму ДО4 визначимо немає чи підграфа H_2, графа F або його частини H_2, де $H_2 = St_F(u_j) \bigcup z$;гомеоморфной $O_n, n = \rho_F(u_j), n > 2$. Якщо "є", то:

$u_j \in Z$;

$F := F \setminus \left(St_F(u_j) \setminus S_1 \right)$;

$z \to M$;

перехід до мітки 2).

Інакше:

2.б) Перевіримо за допомогою алгоритму ДО23 наявність підграфа H_1 графа F гомеоморфного графові $K_{2,3}$ де $H_1 = z \bigcup C$,

$C = C_F(a,b), \quad u_j \in C^0 \setminus \{a,b\}$.

Якщо "є", то

$$u_j \in Z,$$
$$F := F \setminus \left(C\{a,b\}\right),$$
$$z \to M,$$

переходимо до мітки 2).

Інакше: $i` = 1, K := 0,$ перехід до мітки 3).

3) $K := K + 1;$; Якщо $K > |M_{i`}|$, те переходимо до 4). Перевіримо чи маються такі $z_{k`}, z_{k`} \in M_{i`}$ що $z_{k`} \subset \bigcup_{\substack{j \neq k` \\ j=1}}^{m`} Z_j, m` = |M_{i`}|$. Якщо "є", те $M_{i`} := M_{i`} \setminus \{z_{k`}\},$ переходимо до мітки 3). Інакше переходимо до мітки 4).

4) Якщо $i = 1$, те $K = \sum_{i`=1}^{2} |M_{i`}|;$ переходимо до мітки 1).

Якщо $\left(1 < i < |G^0|\right) \wedge \left(\sum_{i=1}^{2} |M_i| > K\right)$, то $K := \sum_{i`=1}^{2} |M_{i`}|$; переходимо до мітки I). Кінець.

Видаємо К. Кінець роботи.

Теорема А. Алгоритм А коректний і має поліноміальну складність для До задовольняючої нерівності: $0 \le K \le 3.$

Доведення. Нехай має місце нерівність $0 \le K \le 3.$

«Алгоритми на графах»

Розглянемо з метою розв'язання наступних задач

1)Побудова основного дерева графа(склета графа);

2)Побудова множини всіх фундаментальних простих циклів графа;

3) Побудова множини всіх циклів графа;

4)обчислення числа досяжності множини вершини графа.

Для задачі №1 використаємо алгоритм «пошук у в глубину графа»

Ідея алгоритму наступна: проходиме по вершинам, відмічаємо ті, що відвідали червоним кольором; повертаємось тим же шляхом та перевіряємо наявність на відповідних вершин.

Остовне дерево для графа є, без вершини 8

Вхід: вешина 1 із маткою $\emptyset$, то ми припиняємо кожній вершині індекс $\emptyset$. Якщо ми її відвідали(вибрали) то ми змінюємо її індекс з нуля на одиницю.

Вибираємо вершину 1 із міткою $\emptyset$, далі дивимось з якими вершинами суміжна наша вершина$1_0(2_0, 3_0, 6_0)$

Припустимо, що ми перейдемо в двійку. Вершина має індекс $\emptyset$ (якби був би якесь ненульове число, то ми не могли б перейти). Переходимо по ребру1,2 в двійку (2_0), та знімаємо мітку на 2_2.

Переходимо в 3_0.

Основне дерево, бо ми пройшли всі вершини, що є в графі.

Алгоритм містить 2 вкладені один в один цикли з параметром bar

, що змінюється від 1 доп де n- кількість вершини

В цьому зовнішньому циклі вибираємо вершину з нульовим індексом(міткою)

У другому циклі параметр bar змінюється від 1 до n,W- вершини індекса; перебирає всі вершини, шо суміжні

Візьмемо 2, вона мала нульову мітку(або в 1,3,11) 3мітки на 1

$2_1 \to 1_2 \quad 1 \to 11_3 \quad 11 \to 3_4$

$3 \to 4_5 \quad 4 \to 5_0 \quad 5_6 \to 6_7$

$6_7 \to 7_8$

Ланцюг із 7 ребер

8,9,10 містить нульові мітки

$7_8 \to 6_7 \quad 6_7 \to 8_9$

$8_9 \to 6_7 \quad 6_7 \to 5_6 \quad 5_6 \to 4_5$

$4_5 \to 9_{10} \quad 9_{10} \to 4_5 \quad 4_5 \to 3_4$

$3_4 \to 11_3 \quad 11_3 \to 10_{11}$

10_{11}- мітка рівня 11. Всього n=11/

$C_1 = L_{2,1,11,3} + l_{10,11,3}$ –не фунтаментальне

Якщо беремо ребро 2,10

$$\left.\begin{array}{r} C_1 = L_{2,1,11,3} + l_2 \\ C_1 = l_1 + l_2 + l_{10,11,3} \end{array}\right\}$$ фундаментальної

$C_0 + C_2 = C_1$

Якщо ми додаємо 2 ребра зі спільними ребрами, то додавання неможливе

$C_4 = l_4 + l_{2,1,11,3,4,9}$- не цикл бо в ньому э діагональ 11,3.

Цикл C_4 можна побудувати якщо берем ребро l_4(11,9)+ простий ланцюг 2,1,11,3,4,9.

Побудуємо цикл C_4з ланцюга 2,1,11,3,4,9 та ребра 9,11. Матимемо заготовку для цикла від якої відріжемо цикл, якщо звернемо увагу на ребро 11,3, що є своєрідною діагональю із заготовку C_4матимемо цикл C_4(простий), який складається з ланцюга 2,11,3,4,9 та ребра C_4. Цей цикл є фундаментальний.

Далі так само побудуємо цикл C_5, C_6, C_7.

У нас 6 фундаментальних циклів.

Будемо вважати цикли C_0, C_1, l_2, C_4, C_5, C_6 множиною всіх фундаментальних циклів .

При побудові цих циклів за допомогою основного дерева ми вибирали ланцюги по дереву та додаткове ребра, що залишаю ці ланцюги.

За допомогою операцій додавання фундаментальних циклів можна побудувати множину всіх можливих циклів з фундаментальних.

Додавання здійснюеться по спільним ребрам, які потім видаляються.

Якщо складаємо C_0 і C_2, тоді ребра(1,10) і (10,11) **???**? (видалились) і ми отримали C_1.

Таким чином можна побудувати розв’язок задаччі 2 та 3.

Але краще побудувати їх в ручну.

Задача №4

Обчислення числа тотожностей

Є вхідний граф G;n=11

Число досяжності є найменшим числом граней (частин площини) на границях яких розміщуються всі вершини графа.

 ля розв’язання задачі обчислення числа досяжності і множини точок площинного графа, яке визначається як найменша кількість граней площинного графа,Є на площинах яких розміщуються всі вершини в графа.

Ідея алгоритму:полягає у побудові множини всіх простих (фундаментальних) циклів та «перебору» по два , три , чотири елементи G та перевіримо чи всі вершини графа.

Наприклад:

Знайдемо число досяжності множини вершини G,

граф G є результатом лінійного синтезу трьох копій графа $K_{2,3}$:

1)Побудуємо основне дерево графа G пошук в глубину вважаючи, що списки суміжних вершин впорядковані по зростанню,

2)До «вільних» (від основного дерева) ребер додаємо ланцюги (заставу), які починаються та закінчуються в кінцевих точках ребер,

3)В непростих циклах проводимо діагоналі, що залишились тобто розіб’ємо їх на прості,

4)Перебором визначаємо мінімальну кількість простих циклів.

«найменший цикл скінченних графів»

Полягає в ототожненні не простих ланцюгах(двох і більше парам) простих графів G_1та G_2 з метою отримання графа G, в якому ?????? «сума» заданих властивостей графів G_1та G_2.

Вироджений випадок лінійного синтезу буде тоді, коли ототожнення по парі простих циклів.

Приклад 1

Приклад 2

Граф G має число досяжності множини вершин 3 «успадковує» суму чисел досяжності графів G_1та G_2. (зменшену на 1) та успадковує мінімальне.

Передбачає виконання всіх різних «склейок» граф G_1та G_2. (із вашого варіанту практичної №2)

Пр. двум простим ланцюгам чи по простих циклах

Приклад 3 (ототожнення по простих циклах)

«Елементи комбінаторного аналізу,(комбінаторики)»

Призначені для переліку підрахунку кількості елементарних комбінаторних конфігурацій, компоненти яких беруть із числа елементів заданої множини.

Найпростіший приклад комбінаторної конфігурації:

1)ланцюжок із компонентів

2)картеж

3)вибірки

4)розміщення

Теорія

Основний принцип комбінаторики

1)$N = N_1 * N_2$, якщо дія 1 не залежить від дії 2 використовуються одночасно

2) $N = N_1 * N_2$, якщо дія 1та дія 2 виконуються в ручний час одна за одною.

Приклади:

1)Скільки існує шестизначних десяткових чисел

2) Скільки існує шестизначних десяткових чисел, у яких друга цифра більша за першу;

Розвязок

Нехай N_i-вибір і тої цифри

N_1=9($\emptyset$ не може бути ??????)

$N_2 = N_3 = N_4 = N_5 = N_6 = 10$

Оскільки вибір цифр не залежить один від одного, то

$N_1 * N_2 * N_3 * N_4 * N_5 * N_6 = 9 * 10^5$- це кількість шестизначних чисел (десяткових).

« існує залежність між вибором другої цифри та першої цифри, тому не працює основний принцип комбінаторики. Тому колективне переберемо варіанти вибору першої пари цифри:

1+2+3+4+5+6+7+8=4*3=36

Інші цифри (3,4,5,6) вибираємо після конструктивного перебору перших двух цифр шестизначного та маємо 10^4-варіантів

Тоді 36*10^4

Приклад 2:

Підрахувати кількість різних трійок із

$\{1,2,3,4,5\}$

(a,b,c)= (a,b,c)

3!-1*2*3=6

Різні трійки мають хоча б одне число яким вони відрізняються

(1,2,3),(1,3,2),(2,3,1),(3,1,2),(2,1,3),(3,2,1)-їх щість штук

Конструктивним способом це всі різні трійки, їх десять, кожна має швсть однакових за порядком складування.

«Задачі комбінаторного аналізу»

Мають на меті перелічити варіанти чи синтезувати їх для заданих комбінаторних конфігурацій (ланцюжки елементів множини).

Тоді за основними принципами комбінаторного аналізу маємо по кількість варіантів розміщення ци 7 тур дорівнюеться

Задача №2

Скільки різних прямокутників в прямокутній таблиці m*?

Де k! – кількість однакових ланцюжків із елементів.

-кількість всії ланцюжків по k –елементів із множини n елементів

Прямокутник визначається двома парами ліній, де кількість різних пар горизонтальних ліній дорівнюватиме C_{m+1}^{z}, а кількість різних пар вертикальних ліній дорівнюватиме C_{n+1}^{z}. Вибір кожної з цих пар не залежить від вибору іншої пари, тому кількість всіх різних прямокутників

Задача №3

Скільки різних ланцюгів можна скласти із літер слова «МАМАНА»

Розв'язок: множина літер слова «МАМАНА» складена з літер $\{2\text{M}, 3\text{A}, 1\text{H}\}$

Різні ланцюги відрізнятимемо за складом (літерами),а не порядком слідування (написання).

1. Множини та операції над ними

Для математики поняття множини дуже важливе. Воно є первісним, тобто йому не дається означення. Можна говорити про множину учнів класу, множину коренів рівняння, множину натуральних чисел. Множини позначаються великими літерами (іноді перераховуючи в фігурних дужках елементи цих множин). Наприклад, множина коренів рівняння $x^3-3x^2+2x=0$ така $\{0, 1, 2\}$. Запис $x \in A$ означає: елемент x належить множині A. Якщо елемент x не належить множині A, то пишуть $x \notin A$. Множина A є підмножиною множини B, якщо кожен елемент x, який належить A, належить і множині B; позначається це так $A \subset B$. Множина, яка не містить жодного елемента називається порожньою і позначається $\varnothing$.

У багатьох розділах математики і, зокрема, в теорії ймовірностей розглядаються лише множини, що є підмножинами однієї і тієї ж множини, яку називають універсальною множиною. (В планіметрії, наприклад, в ролі такої множини виступає множина усіх точок площини.) Дотримуючись традиції, прийнятої в теорії ймовірностей, будемо позначати універсальну множину великою грецькою літерою Ω (омега).

Над множинами можна виконувати певні операції. Нехай A і B дві множини.

Означення 1. *Об'єднанням $A \cup B$ множин A і B називається множина всіх тих і тільки тих елементів, які належать принаймні одній з множин A або B:*

$$A \cup B = \{x \mid x \in A \text{ або } x \in B\}.$$

Означення 2. *Перетином $A \cap B$ множин A і B називається множина всіх тих і тільки тих елементів, які належать як A, так і B:*

$$A \cap B = \{x \mid x \in A \text{ і } x \in B\}.$$

Означення 3. *Різницею $A \backslash B$ множин A і B називається множина тих і тільки тих елементів, які належать A і не належать B:*

$$A \backslash B = \{x \mid x \in A \text{ і } x \notin B\}.$$

Означення 4. *Нехай $A \subset \Omega$. Доповненням $\overline{A}$ до множини A називається множина $\Omega \backslash A$:*

$$\overline{A} = \Omega \backslash A = \{x \mid x \in \Omega \text{ і } x \notin A\}.$$

Означення 5. *Симетричною різницею $A \circ B$ множин A і B називається множина тих і тільки тих елементів, які належать A або B і не належать $A \cap B$:*

$$A \circ B = \{x \mid (x \in A \text{ або } x \in B) \text{ і } x \notin A \cap B\}.$$

Корисно мати схематичне зображення універсальної множини і її підмножин. Це можна зробити за допомогою так званих діаграм Венна. Часто універсальну множину зображають прямокутником, а підмножини - кругами.

Проілюструємо операції над множинами за допомогою діаграм Венна (рис.1):

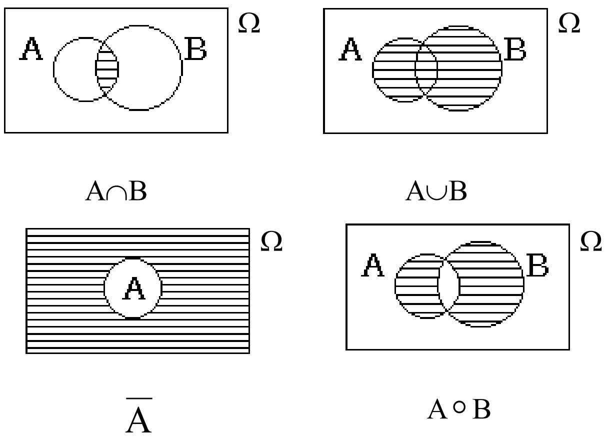


рис. 1

Приклад 1. Нехай $\Omega=\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$. A - множина парних чисел з Ω, B - множина тих чисел Ω, які кратні 3. Тоді

$$A=\{2, 4, 6, 8, 10\},\ B=\{3, 6, 9\},\ A \cup B=\{2, 3, 4, 6, 8, 9, 10\},\ A \cap B=\{6\},$$

$$A \backslash B=\{2, 4, 8, 10\},\ \overline{A}=\{1, 3, 5, 7, 9\},\ \overline{B}=\{1, 2, 4, 5, 7, 8, 10\}.$$

2. Правила комбінаторики

Розпочнемо знайомство з комбінаторикою з двох основних правил, які можна вважати за аксіоми комбінаторики. Спочатку проілюструємо ці правила на прикладах.

Приклад 1. У кімнаті відкрито чотири вікна та двоє дверей. Скількома способами можна потрапити до кімнати?

До кімнати можна потрапити або через вікно (чотирма способами) або через двері (двома способами). Отже, до кімнати можна потрапити 4+2=6 способами.

І взагалі, *якщо деякий об'єкт A можна вибрати m способами, а другий об'єкт B n-способами, причому ніякий вибір A не збігається з жодним з виборів B, то один з об'єктів A або B можна вибрати n+m способами*. Це і є перше правило комбінаторики- **правило суми**.

Приклад 2. З пункту A до пункту B можна дістатися пароплавом, поїздом та літаком; з пункту B до пункту C- автобусом та поїздом. Скількома способами можна здійснити подорож з пункту A до пункту C через пункт B (рис.2).

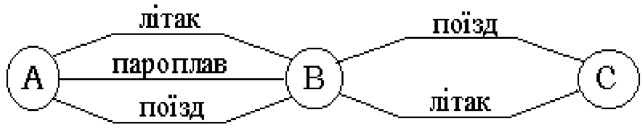


рис.2

Якщо ми оберемо один з трьох можливих способів подорожі з пункту A до пункту B, то будемо мати два можливих способи подорожі з пункту B до пункту C. Отже, різних шляхів з пункту A до пункту C буде $3 \cdot 2 = 6$.

І взагалі, нехай із множини, яка містить m елементів, вибирається один елемент і незалежно з множини, яка містить n елементів також вибирається один елемент. Питається: скільки різних пар елементів при цьому можна утворити?

Відповідь на поставлене питання дає таблиця:

$$\left.\begin{array}{cccc} a_1b_1; & a_1b_2; & \ldots; & a_1b_n \\ a_2b_1; & a_2b_2; & \ldots; & a_2b_n \\ \ldots & \ldots & \ldots & \ldots \\ \multicolumn{4}{c}{\underbrace{a_mb_1; \quad a_mb_2; \quad \ldots; \quad a_mb_n}_{n \text{ пар у кожному рядку}}} \end{array}\right\} m \text{ рядків}$$

Таким чином, загальне число усіх можливих пар буде дорівнювати $m \cdot n$. Це і є **правило добутку**, яке формулюється так: *якщо деякий об'єкт A можна вибрати m способами, а деякий другий об'єкт B можна вибрати n способами, то пару об'єктів A і B у вказаному порядку можна вибрати* $m \cdot n$ *способами.*

Узагальнимо ці правила. Для цього використаємо таке позначення: якщо X - це скінченна множина, то символ |X| - кількість елементів множини X.

Правило суми. *Нехай множини* X_1, X_2, ...,X_n *попарно не перетинаються. Тоді* $|X_1+X_2+\ldots+X_n|=|X_1|+|X_2|+\ldots+|X_n|$.

Правило добутку. *Нехай* X_1, X_2, ...,X_n *- довільні множини. Тоді* $|X_1 \times X_2 \times \ldots \times X_n|=|X_1| \times |X_2| \times \ldots \times |X_n|$, де знак $\times$ - знак декартового добутку множин.

Ці правила легко доводяться методом математичної індукції, якщо їх вважати вірними для двох множин.

Приклад 3. Скільки різних дільників має число 2310?

Розв'язання. Оскільки $2310=2 \cdot 3 \cdot 5 \cdot 7 \cdot 11$, то число d буде дільником 2310, якщо d - добуток чисел з набору 2, 3, 5, 7, 11. Будемо складати всі можливі добутки з цих чисел такими міркуваннями: перше число може входити до добутку або не входити, друге число також може входити або не входити до добутку, аналогічна ситуація і з іншими числами. Тому різних дільників числа 2310, які є добутком чисел з набору 2, 3, 5, 7, 11, буде 2^5. З цього числа виключимо випадок, коли d=0 (жодне число з вказаного набору чисел не входить до d), і додамо випадок, коли d=1. Отже число 2310 має 32 різних дільника.

3. Вибірки

Під вибіркою будемо розуміти сукупність деяких елементів, вибраних із заданої множини певним чином. Вдалим прикладом вибірки може бути довільний, в тому числі і беззмістовний, текст, введений до комп'ютера з клавіатури і відображений на екрані. Заданою множиною - є множина клавіш клавіатури, вибіркою - довільна послідовність символів, введених з клавіатури. З цього прикладу зрозуміло, що до складу вибірки можуть входити і однакові елементи. Вибірки, які містять однакові елементи, будемо називати вибірками з повтореннями, в іншому випадку - вибірками без повторень. Очевидно, що кількість елементів вибірки з повтореннями може бути більшою від кількості елементів вихідної множини.

Вибірки бувають впорядкованими і невпорядкованими. Наприклад, щоб комп'ютер виконав ту чи іншу дію, ми повинні з клавіатури безпомилково ввести деяку команду. Навіть, якщо поміняти місцями два довільні символи команди, то очікуваного результату ми не отримаємо. Така команда буде прикладом впорядкованої вибірки. А, наприклад, коли беруться книжки для роботи з книжкової полиці, то порядок вибору цих книжок ролі не грає. Ця вибірка з книжок і буде прикладом невпорядкованої вибірки.

Отже, маємо:

вибірки

↙ ↘

впорядковані невпорядковані

↙ ↘ ↙ ↘

без повторень з повтореннями без повторень з повтореннями

При розв'язуванні задач потрібно вміти знаходити кількість вибірок того чи іншого виду, найпоширеніші з яких названо розміщеннями, перестановками, комбінаціями.

Нехай є множина B, яка складається з n елементів.

Означення 1. *Впорядковані k-елементні вибірки без повторень множини B називаються розміщеннями без повторень з n елементів по k елементів.*

Число розміщень без повторень з n елементів по k елементів позначається символом A_n^k, який походить від початкової букви французького слова *arrangement*, що означає "розміщення".

Означення 2. *Впорядковані k-елементні вибірки з повтореннями множини B називаються розміщеннями з повтореннями з n елементів по k елементів.*

Число розміщень з повтореннями з n елементів по k елементів позначається символом $\overline{A}_n^k$.

Означення 3. *Впорядковані n-елементні вибірки без повторень множини B називаються перестановками без повторень з n елементів.*

Число перестановок без повторень з n елементів позначається символом P_n, який походить від початкової букви французького слова *permutation*, що означає "перестановка".

Означення 4. *Невпорядковані k-елементні вибірки без повторень множини B називаються комбінаціями без повторень з n елементів по k елементів.*

Число комбінацій без повторень з *n* елементів по *k* елементів позначається символом C_n^k, який походить від початкової букви французького слова *combination*, що означає “комбінація”.

Означення 5. *Невпорядковані k-елементні вибірки з повтореннями множини В називаються комбінаціями з повтореннями з n елементів по k елементів.*

Число комбінацій з повтореннями з *n* елементів по *k* елементів позначається символом $\overline{\mathrm{C}}_n^k$.

Для числа таких вибірок виведені формули, якими користуються при розв'язанні різних комбінаторних задач.

Приклад 1. Нехай множина В складається з трьох елементів В={*a*, *b*, *c*}. Виписати:

а) двоелементні розміщення без повторень,

б) двоелементні розміщення з повтореннями,

в) двоелементні комбінації без повторень,

г) двоелементні комбінації з повтореннями,

д) перестановки без повторень.

Підрахувати кількість вибірок кожного виду.

Розв’язання.

а) *<a, b>, <b, c>, <a, c>, <b, a>, <c, b>, <c, a>*. $\mathrm{A}_3^2=6$;

б) *<a, b>, <b, c>, <a, c>, <b, a>, <c, b>, <c, a>,<a, a>, <b, b>, <c, c>*. $\overline{\mathrm{A}}_3^2=9$;

в) *<a, b>, <b, c>, <a, c>*. $\mathrm{C}_3^2=3$;

г) *<a, b>, <b, c>, <a, c>,<a, a>, <b, b>, <c, c>*. $\overline{\mathrm{C}}_3^2=6$;

д) *<a, b, c>, <b, c, a>, <c, a, b>, <a, c, b>, <b, a, c>, <c, b, a>*. $P_3=6$.

Як виписати всі вибірки того чи іншого типу, щоб не помилитися? Існує зручний спосіб, при якому виключаються пропуски,- побудова так званого дерева. Наприклад, випишемо всі перестановки без повторень. Початкову точку - вершину дерева - позначимо буквою О. Від неї будемо рухатись вниз всіма можливими шляхами, як показано на рис.3.

```
            0
         /  |  \                       0
      a     b     c                 /  |  \
      ^     ^     ^              a     b     c
    b  c  a  c  a  b             ^     ^     ^
    |  |  |  |  |  |           b  c  a  c  a  b
```

c b c a b a

рис. 3 рис. 4

Тепер випишемо всі шляхи-гілки нашого дерева. Отримаємо всі можливі перестановки множини В: <a, b, c>, <b, c, a>, <c, a, b>, <a, c, b>, <b, a, c>, <c, b, a>.

На рис.4 наведено приклад дерева для знаходження всіх розміщень без повторень з 3 елементів по 2.

4. Розміщення

Теорема 1. *Має місце рівність*

(1) $$\mathrm{A}_n^k = n(n-1)(n-2)...(n-k+1).$$

Доведення.

Для доведення необхідно визначити скільки різних впорядкованих k-елементних вибірок без повторень можна утворити з n елементів множини В. Нехай $\langle b_1, b_2,..., b_k\rangle$- довільна k- елементна вибірка множини В. Елемент b_1 можемо вибрати n способами. Оскільки елемент b_2 повинен бути відмінним від b_1, то b_2 - можна вибрати (n-1) способами. Аналогічно b_3 - (n-2) способами,..., b_k - (n-(k-1)) способами. Тоді за правилом добутку кількість k- елементних вибірок без повторень множини В дорівнює

$$\mathrm{A}_n^k = n(n-1)(n-2)...(n-k+1).$$

Для обчислення числа розміщень без повторень зручно користуватися іншою формулою. Щоб її отримати помножимо чисельник і знаменник правої частини формули (1) на вираз

$$(n-k)\cdot(n-(k+1))\cdot(n-(k+2))\cdot\ldots\cdot 3\cdot 2\cdot 1 .$$

Маємо:

$$\mathrm{A}_n^k = \frac{n\cdot(n-1)\cdot(n-2)\cdot\ldots\cdot(n-(k-1))\cdot(n-k)\cdot(n-(k+1))\cdot\ldots\cdot 3\cdot 2\cdot 1}{(n-k)\cdot(n-(k+1))\cdot\ldots\cdot 3\cdot 2\cdot 1} .$$

Математики ввели спеціальну назву для добутку

$$n\cdot(n-1)\cdot(n-2)\cdot\ldots\cdot 3\cdot 2\cdot 1 .$$

Такий добуток називається факторіалом числа n і позначається символом $n!$.

Отже

$$\mathrm{A}_n^k = \frac{n!}{(n-k)!} .$$

Теорема 2. *Має місце формула*

$$\overline{\mathrm{A}}_n^k = n^k.$$

Доведення.

Аналогічно до попереднього, якщо <b_1, b_2, ..., b_k> довільна впорядкована k- елементна вибірка з повтореннями множини B, то елемент b_1 можна вибрати n способами, b_2 - теж n способами, ..., b_k - n способами, бо елементи можуть повторюватися. Отже за правилом добутку кількість шуканих вибірок дорівнює добутку k однакових множників, кожен з яких дорівнює n. Тому

$$\overline{\mathrm{A}}_n^k = n^k.$$

Приклад 1. Скількома способами можна з 30 учнів класу обрати старосту, його заступника і редактора стінгазети.

Розв'язання. В цій задачі необхідно знайти число розміщень без повторень з 30 елементів по 3, оскільки тут має значення не лише, хто буде обраний до керуючої трійки, а й те, як будуть розподілені посади між обраними учнями Тому відповідь знаходиться за формулою

$$\mathrm{A}_{30}^3 = \frac{30!}{(30-3)!} = 30 \cdot 29 \cdot 28 = 24360.$$

Приклад 2. Скільки різних тризначних чисел можна скласти з цифр 1, 2, 3, 4, 6, 8.

Розв'язання. Відповідь знаходимо за формулою

$$\overline{\mathrm{A}}_6^3 = 6^3 = 216.$$

5. Перестановки

Теорема 1. *Має місце формула*

$$P_n = n!.$$

Доведення.

З означення перестановок зрозуміло, що перестановки з n елементів- це розміщення без повторень з n елементів по n елементів. Отже

$$\mathrm{P}_n = \mathrm{A}_n^n = 1 \cdot 2 \cdot 3 \cdot \ldots \cdot \mathrm{n} = n!$$

Приклад 1. Одного разу 10 друзів зайшли до кав'ярні. Господар кав'ярні запропонував їм приходити до нього щодня і кожного разу сідати за той самий стіл по-іншому; після того, як усі способи розміщень будуть вичерпані, їх пригощатимуть у кав'ярні безкоштовно. Коли настане цей день?

Розв'язання. Число різних способів розміщення 10 чоловік за столом дорівнює числу перестановок з 10 елементів. Оскільки

$$\mathrm{P}_{10} = 10! = 3628800,$$

то цей день настане через 3628800 дня або через 9942 роки.

Нехай є множина B={b_1, b_2, ..., b_n}. Розглянемо лише ті k-елементні розміщення з повтореннями множини B, які задовольняють такій умові: кожна вибірка-розміщення містить k_1 елементів b_1, k_2 елементів b_2, ..., k_n елементів b_n, причому $k_1+k_2+\ldots+k_n=k$. Зрозуміло, що такі вибірки одна від іншої будуть відрізнятися лише порядком слідування елементів. Їх називають перестановками з повтореннями з k елементів і позначають символом $\mathrm{P}_k(k_1, k_2, \ldots, k_n)$.

Теорема 2. *Має місце формула*

$$P_k(k_1, k_2,..., k_n)=\frac{k!}{k_1!\cdot k_2!\cdot\ldots\cdot k_n!}.$$

Доведення.

Розглянемо довільну k- елементну вибірку-перестановку, яка містить k_1 елементів b_1, k_2 елементів b_2, ..., k_n елементів b_n, причому $k_1+k_2+...+k_n=k$ (наприклад $<\underbrace{b_1b_1\ldots b_1}_{k_1}\underbrace{b_2b_2\ldots b_2}_{k_2}\ldots\underbrace{b_nb_n\ldots b_n}_{k_n}>$).

Підрахуємо скільки нових перестановок з повтореннями з неї можна отримати.

Спочатку знайдемо скількома способами можна переставляти її елементи, щоб перестановка не змінилася. Оскільки елементи b_1 можна переставляти $k_1!$ способами, елементи b_2 - $k_2!$ способами, ..., елементи b_n - $k_n!$ способами, то за правилом добутку в кожній перестановці з повтореннями можна переставляти елементи (не змінюючи перестановку) $k_1!\cdot k_2!\cdot\ldots\cdot k_n!$ способами.

Всього ж елементи розглядуваної вибірки можна переставляти $k!$ способами.

Отже, число різних перестановок з повтореннями з k елементів менше $k!$ в $k_1!\cdot k_2!\cdot\ldots\cdot k_n!$ раз. Тому

$$\mathrm{P}_k(k_1, k_2,..., k_n)=\frac{k!}{k_1!\cdot k_2!\cdot\ldots\cdot k_n!}.$$

Приклад 2. Скільки різних слів, у тому числі і беззмістовних, можна дістати, якщо переставляти букви в слові математика.

Розв'язання. Очевидно цим числом є

$$\mathrm{P}_{10}(2, 3, 2, 1, 1, 1)=\frac{10!}{2!\cdot 3!\cdot 2!}=151200.$$

6. Комбінації

Розглянемо множину B={a, b, c}. Як було встановлено в прикладі 3.1, комбінаціями без повторень з 3-х елементів по 2 і розміщеннями без повторень з 3-х елементів по 2 є такі вибірки:

комбінації: <*a*, *b*> <*b*, *c*> <*a*, *c*>

/ \ / \ / \

розміщення: <*a*, *b*> <*b*, *a*> <*b*, *c*> <*c*, *b*> <*a*, *c*> <*c*, *a*>

Неважко помітити, що з кожної комбінації, змінюючи порядок слідування її елементів, ми отримуємо $\mathrm{P}_2=2$ вибірок, які є розміщеннями без повторень з 3-х елементів по 2. Отже , маємо залежність $\mathrm{A}_3^2=\mathrm{C}_3^2\cdot 2$. А звідси, уміючи обчислювати A_3^2, знаходимо $\mathrm{C}_3^2=\frac{\mathrm{A}_3^2}{2}=\frac{6}{2}=3$.

Поширимо наші міркування на загальний випадок.

Теорема 1. *Має місце формула*

$$C_n^k = \frac{n!}{k! \cdot (n-k)!}.$$

Доведення.

Нехай маємо *n*-елементну множину В. Ця множина має C_n^k комбінацій. Розглянемо одну з них. Змінюючи порядок слідування елементів цієї комбінації, ми отримаємо *k*! впорядкованих вибірок, які є розміщеннями з *n* елементів по *k* елементів. Аналогічно розглядаючи всі інші комбінації, приходимо до висновку, що число розміщень з *n* елементів по *k* для множини В дорівнює $A_n^k = k!$ C_n^k. Звідси маємо

$$C_n^k = \frac{A_n^k}{k!} = \frac{n!}{k! \cdot (n-k)!}.$$

Приклад 1. Скількома способами можна з 30 учнів класу обрати делегацію у складі трьох чоловік.

Розв'язання. В цьому випадку, хоча умова задачі дуже схожа з умовою задачі 4.1, необхідно знайти число комбінацій без повторень з 30 елементів по 3 Тому що, в даному випадку має значення лише те, кого оберуть до складу делегації. Маємо відповідь

$$C_{30}^3 = \frac{30!}{3! \cdot (30-3)!} = 4060.$$

Приклад 2. З дитинства ми пам'ятаємо, що в чарівному мішку Діда Мороза подарунки ніколи не кінчаються. Нехай новорічний мішок містить апельсинки, горішки і цукерки. Кожній дитині Дід Мороз принесе подарунок, який містить чотири гостинці. Скільки різних подарунків може скласти Дід Мороз?

Розв'язання. Переглядаючи різні подарунки, домовимося кожен гостинець подарунку відмічати окремо: спочатку наявність апельсинок, потім горішків, потім цукерок. Щоб не писати назви гостинців, будемо один вид гостинця відділяти від іншого рискою (перегородкою), а наявність гостинця відмічати плюсом. Наприклад,

<+ | + | + +> - цей запис означає, що подарунок складається з однієї апельсинки, одного горішка та двох цукерок,

<+ | | + + +> - цьому ж запису відповідає подарунок, який містить одну апельсинку, жодного горішка та три цукерки.

Отже кожному подарунку відповідає своя шестиелементна послідовність з 2-х рисок і 4-х плюсів, а кожній такій послідовності - свій подарунок. Тому різних подарунків стільки, скільки існує різних послідовностей з 2-х рисок і 4-х плюсів. А цих послідовностей стільки, скількома способами можна вибрати два елемента (місце положення рисок) з 4+2=6 елементів, тобто C_{4+2}^2.

Легко помітити, що рисок на одну менше ніж видів гостинців, тобто 3-1=2.

Отже, число різних подарунків, які складено з 3-х видів гостинців по 4-ри гостинці, тобто число комбінацій з повторенням з 3-х елементів по 4, ми знаходимо такими міркуваннями

$$\overline{C}_3^4 = C_{4+(3-1)}^{3-1} = C_{4+3-1}^{3-1} = C_6^2 = 15.$$

Ці міркування дозволяють встановити таке твердження.

Теорема 2. *Має місце формула*

$$\overline{C}_n^k = C_{k+n-1}^{n-1}.$$

Доведення.

Нехай B={b_1, b_2, ..., b_n}. Підрахуємо для цієї множини число $\overline{C}_n^k$.

Оскільки в комбінаціях з повторенням порядок слідування елементів не береться до уваги, то любу комбінацію з повторенням з n елементів по k можна представити так:

(1) $$<\underbrace{b_1 b_1 \ldots b_1}_{k_1}\underbrace{b_2 b_2 \ldots b_2}_{k_2}\ldots\underbrace{b_n b_n \ldots b_n}_{k_n}>,$$

де $k_1+k_2+...+k_n=k$ (тобто спочатку записати всі елементи b_1, потім всі елементи b_2 і т.д.).

Цей запис можна ще спростити, якщо групи однакових елементів відділяти рисками, а наявність кожного елемента відмічати плюсом. Наприклад для вибірки (1) матимемо

(2) $$\underbrace{+\,+\ldots+}_{k_1}|\underbrace{+\,+\ldots+}_{k_2}|\ldots|\underbrace{+\,+\ldots+}_{k_n}.$$

Отже для кожної вибірки виду (1) можна записати свою послідовність з k+(n-1) плюсів і рисок. А кожній послідовності виду (2) відповідає певна вибірка. Тому комбінацій з повтореннями з n елементів по k стільки, скільки існує послідовностей виду (2). Послідовностей же виду (2) стільки, скількома способами можна вибрати (n-1) елемент з k+(n-1) елементів.

Тобто задача звелася до підрахунку числа комбінацій без повторень з k+(n-1) елементів по (n-1) елемент. Таким чином

$$\overline{C}_n^k = C_{k+n-1}^{n-1}.$$

Приклад 4. Скількома способами можна купити 8 тістечок у кондитерській, де є 6 різних їх сортів.

Розв'язання. Відповідь знаходимо за формулою

$$\overline{C}_6^8 = C_{13}^8 = 1287.$$

7. Біном Ньютона. Властивості біноміальних коефіцієнтів

Числа C_n^k часто зустрічаються в різних розділах математики. Вони давно відомі і мають багато важливих властивостей. Розглянемо деякі з них.

1. $C_n^k = C_n^{n-k}$ (1)

2. $C_n^k = C_{n-1}^{k-1} + C_{n-1}^k$ (2)

Ці властивості можна довести використовуючи співвідношення для обчислення числа комбінацій без повторень. Але корисно їх довести чисто комбінаторним шляхом. Спочатку доведемо (1).

Доведення.

Розглянемо квадратну сітку розмірами $k \times m$ (“шахове місто”, яке складається з $k \times m$ прямокутних кварталів, поділених m-1 “горизонтальними” і k-1 “вертикальними” вулицями (рис.5)). Яке число різних найкоротших доріг на цій сітці, які ведуть з лівого нижнього кута (точки (0,0)) в правий верхній кут (точку (k, m))?

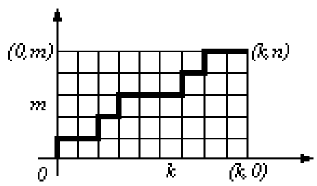


рис. 5

Кожен найкоротший шлях з точки (0, 0) в точку (k, m) складається з $k + m$ відрізків, причому серед них є k горизонтальних і m вертикальних відрізків. Тому загальне число шляхів дорівнює числу способів, якими з $k + m$ відрізків можна вибрати m вертикальних відрізків, тобто C_{k+m}^{m}.

Можна було б розглядати число способів вибору не m вертикальних, а k горизонтальних відрізків, і ми одержали б тоді відповідь C_{k+m}^{k}. Отже ми встановили рівність $C_{k+m}^{k} = C_{k+m}^{m}$. Позначимо $k + m = n$. Тоді

$$C_n^k = C_n^{n-k}.$$

Доведемо тотожність (2).

Доведення. Число найкоротших шляхів з точки (0, 0) в точку A(k, n-k) дорівнює $C_{k+(n-k)}^{k} = C_n^k$ (рис. 6). Всі такі шляхи можна поділити на дві групи: шляхи, які проходять через точку A_1(k-1, n-k) (число їх дорівнює $C_{(k-1)+(n-k)}^{k-1} = C_{n-1}^{k-1}$), і шляхи, які проходять через точку A_2(k, n-k-1) (число їх дорівнює $C_{(n-k-1)+k}^{k} = C_{n-1}^{k}$). Тоді за правилом суми

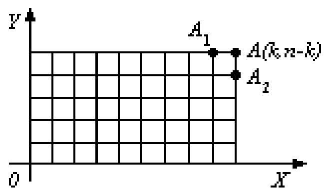


рис. 6

$$C_n^k = C_{n-1}^{k-1} + C_{n-1}^{k}.$$

Тотожність (2) дозволяє послідовно виписувати числа C_n^k у вигляді трикутної таблиці, яка називається *трикутником Паскаля*:

$$\begin{array}{ccccccccccc}
 & & & & 1 & & 1 & & & & \\
 & & & 1 & & 2 & & 1 & & & \\
 & & 1 & & 3 & & 3 & & 1 & & \\
 & 1 & & 4 & & 6 & & 4 & & 1 & \\
1 & & 5 & & 10 & & 10 & & 5 & & 1
\end{array}$$

При різноманітних перетвореннях, в яких використовують числа C_n^k, вважається що C_n^{-1}:=0, C_n^{n+1}:=0.

Числа C_n^k *в математиці називають біноміальними коефіцієнтами*, бо вони використовуються для запису формули бінома Ньютона, тобто

(3) $$(a + b)^n = \sum_{k=0}^{n} C_n^k a^k b^{n-k}$$

В n-му рядку трикутника Паскаля стоять коефіцієнти розкладу $(a + b)^n$, причому кожен коефіцієнт, крім крайніх, дорівнює сумі відповідних коефіцієнтів з попереднього рядка.

Формула бінома Ньютона настільки важлива, що корисно навести декілька її доведень.

Доведення 1 (комбінаторне).

Запишемо ліву частину (3) у вигляді добутку n однакових множників

$$(a+b)^n = \underbrace{(a+b)(a+b)\ldots(a+b)}_{\text{n множників виду (a+b)}}.$$

Розкриємо дужки. Це можна зробити так: виберемо довільний доданок d_1 з першого множника, помножимо його на довільний доданок d_2 з другого множника, . . ., помножимо на довільний доданок d_n з n-го множника. Після розкриття дужок одержимо суму доданків виду $d_1d_2d_3 \ldots d_n$, де кожен з множників дорівнює або a, або b. Підрахуємо кількість всіх доданків. Кожен з множників d_i ($i = 1, 2, \ldots, n$) можна вибрати одним з двох способів, тому застосувавши правило добутку одержимо число доданків рівне 2^n. Кожне d_i ($i = 1, 2, \ldots, n$) дорівнює a, або b, тому доданки можна записати і у вигляді

(4) $$a^k b^{n-k},$$

де $k = 0, 1, \ldots, n$. Оскільки k може приймати значення від 0 до n, то виразів виду (4) буде всього n+1. А доданків після розкриття дужок утворилося 2^n. Це означає, що не всі доданки різні. Підрахуємо скільки утворилося доданків виду (4) для конкретного k. Очевидно їх стільки, скількома способами з n множників $d_1d_2d_3 \ldots d_n$ можна вибрати k множників, які дорівнюють a, тобто C_n^k. А оскільки k може приймати значення від 0 до n, то

$$(a+b)^n = \sum_{k=0}^{n} C_n^k a^k b^{n-k}.$$

Доведення 2 (методом математичної індукції).

1. Для n=1 формула (3) має місце: $(a + b)^1 = a + b = C_1^0 a + C_1^1 b$.

2. Припустимо, що формула (3) справедлива і для n=m, тобто

$$(a+b)^m = \sum_{k=0}^{m} C_m^k a^k b^{m-k}.$$

3. Доведемо, що рівність (3) має місце і для n=m +1. Справді

$$(a+b)^{m+1} = (a+b)^m(a+b) = \left(\sum_{k=0}^{m} C_m^k a^k b^{m-k}\right)(a+b) = \sum_{k=0}^{m} C_m^k a^k b^{m-k}\,(a+b) =$$

$$= \sum_{k=0}^{m} C_m^k a^{k+1} b^{m-k} + \sum_{k=0}^{m} C_m^k a^k b^{m+1-k} = \{\, k+1 = k_1,\ 1 \le k_1 \le m+1 \,\} =$$

$$= \sum_{k_1=1}^{m+1} C_m^{k_1-1} a^{k_1} b^{m+1-k_1} + \sum_{k=0}^{m+1} C_m^k a^k b^{m+1-k} = \sum_{k=1}^{m+1} C_m^{k-1} a^k b^{m+1-k} + \sum_{k=0}^{m+1} C_m^k a^k b^{m+1-k} =$$

$$=\sum_{k=0}^{m+1}\left(C_m^{k-1}+C_m^k\right)a^k b^{m+1-k}=\sum_{k=0}^{m+1}C_{m+1}^k a^k b^{m+1-k}\ .$$

Доведення 3 (аналітичне).

Для такого доведення використовується формула Тейлора для многочленів:

$$f(x)=\sum_{k=0}^{n}\frac{f^{(k)}(0)}{k!}x^k\ ,$$

де $f(x)$ - многочлен n-го степеня.

Застосуємо її до многочлена $f(x) = (1 + x)^n$. Маємо

$$f(x) = (1 + x)^n = \sum_{k=0}^{n} a_k x^k\ .$$

Знайдемо a_k :

$$f^{(n)}(x) = n(n-1)...(n-k+1)(1+x)^{n-k},\ f^{(k)}(0) = n(n-1)...(n-k+1),$$

$$a_k = \frac{f^{(k)}(0)}{k!} = \frac{n(n-1)\ldots(n-k+1)}{k!} = C_n^k\ .$$

Отже

$$(1 + x)^n = \sum_{k=0}^{n} C_n^k x^k\ .$$

Покладемо $x = \dfrac{a}{b}$, тоді

$$\left(1+\frac{a}{b}\right)^n = \sum_{k=0}^{n} C_n^k \left(\frac{a}{b}\right)^k\ ,$$

$$(a + b)^n = b^n\sum_{k=0}^{n} C_n^k a^k b^{-k} = \sum_{k=0}^{n} C_n^k a^k b^{n-k}\ .$$

8. Відображення: вступні зауваження

В математиці поняття відображення, як і поняття множини, належить до числа основних понять. Тому ми не даємо йому формального означення, а лише пояснюємо зміст цього поняття.

Відображення завжди визначається на деякій множині, яка називається його “областю визначення”. Відображення f ставить кожному елементу x своєї області визначення D_f цілком певний об’єкт

$$y=f(x).$$

Множина всіх об’єктів y, що відповідають всім можливим $x\in D_f$, називається множиною значень відображення і позначається E_f.

Необхідно чітко розрізняти таку термінологію:

“відображення на множину”,

“відображення в множину”.

Відображення множини N на множину M - це відображення з областю визначення N і областю значень M. Поняття *відображення множини N в множину M* ширше - *це відображення з областю визначення N і областю значень, яка є підмножиною множини M* (рис.7).

Практична робота №1

ТЕМА: Визначення умовно незалежних подіі.

МЕТА: Оволодіти методикою визначення ймовірності довільної події по стійкості накопиченої частоти.

ТЕОРІЯ: Для визначення ймовірності події можливо використати наступний "частотний" метод, що грунтується на стійкості послідовності значень появи події. Згідно цього методу слід провести серію дослідів однієі розмірності, в кожному з яких підраховують N_i кількість тих випадків, коли настає подія, де і-номер досліду. Накопичена частота V_n обчислюється за такою формулою.

$$V_n = \sum_{i=1}^{n} \frac{N_i}{i \bullet d}$$

Отримані числа записуються в таблицю.

Отриману послідовність $\{ V_n \}$ дослідимо на предмет наявності властивості стабільності. Ця властивість полягає в тому, що починаючи з номера N для всіх n>N матиме місце нерівність:

$$| V_n - P | < E \quad , \quad E=0{,}0001$$

Якщо Е-нескінченно мала величена, то маємо рівність.

$$P = \lim_{n \to \infty} V_n$$

n → ∞ -нескінченість

де n-номер серіі досліду, Vn-накопичена частота появи букви після n-тоі серіі.

Задачі:

1.Визначити ймовірність появи літери, що має порядковий номер, який співпадає із порядковим номером вашого прізвища в журналі групи:

а) вручну для тексту на укр.мові розміром 5кб.

б) програмно для тексту на укр.мові розміром >50кб.

2.Визначити ймовірність появи складу, який розпочинається літерою з пункту 1 та умовну (байесівську) залежнсть між ними.

3.Побудувати таблицю ймовірностей всіх літер алфавіту.

Приклад..Практичне обчислення ймовірності події присвячене розв’язання задачі №1, для «О», «о» :

1) беремо текст із частини газетної української статті
2) Складаємо таблицю для порції на 100 літер

№порції	1	2	3		47	48	49	50
Кількість в порції «О» , «о»	4	5	6		Числа без зростання при $\epsilon = 0{,}002$			
Накопичена_частота	$\frac{4}{100}$	$\frac{4+5}{100}$	$\frac{4+5+6}{300}$				А=0,058	

3)Висновок щодо стійкості робимо розглядаючи числа (№№50,49,48,47). Якшо в кінці таблиці(50,29,28..) є ближчими до числа А і відзначаються від цього (А) на одну тисячну. А=0,058.

За імовірність появи літери візьмемо середнє арифметичне цих послідовних чотирьох чисел, на якій спостерігається стійкість.

Хід розв'язання задачі 1 для літери О чи о матиме наступний вигляд:

5) Відкриваемо файл β
6) Поки не кінець файла β виконувати:

Зчитуємо символ m.

Якщр це літера укр. Алфавіту чи пробел то збільшуємо на 1 значення лічильника 0, інакше переходимо до кінця циклу читання файлу β.

Якшо ASCI код символа m співпадає із кодом літери О чи о, то збільшуємо на 1 значення лічильника 1 та переходимо до кінця циклу читання файлу β, інакше переходимо до кінця циклу читання файлу β..

3) Кінець циклу читання файлу β

Практична робота №2

ТЕМА: Подання графа для обробки за допомогою комп'ютера

МЕТА: Вивчити способи подання графів як інформаційних об'єктів для комп'ютерної обробки.

Завдання: Вибрати згідно вашого номера в списку групи відповідний граф та виконати наступне:

1) вручну подати граф за всіма способами;
2) запрограмувати подання довільного скінченого графа.

ТЕОРІЯ:

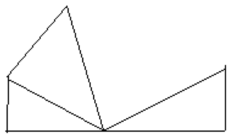
1) Представлення графів в пам'яті залежить від структури данних,які допускае алгоритмична мова та типу ЕОМ.

a) Представлення за допомогою матриці суміжності,порядок якої співподае із числом вершин,

де елемент (i-j) –й дорівнюе 1, якщо i-та вершина суміжна із j-ою вершиною, та (i-j)-й елемент рівний 0 в протележному випадку.

Для графів із великою кількістью дуг це досить компактне представлення, а для графів із невеликім числом дугматимо досить розріджену матрицю. Наведемо приклади таких представлень для неорінтовного графа: G1 та орграфа G2:

x3



G1: x2 x5

010100

01100

x1 x4 x6 A(G1)= 0100

011

01

0

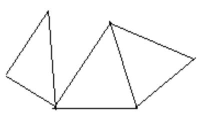
011000

x2 000000

x4 010110

x6 A(G2)= 000010

G2: x1 000001



000100

x3 x5

б) Представлення за допомогою матриці інцедентностей визначае граф однозначно бо мае порядок n x m , де n-кількість вершин,а m-кількість ребер; елементи матриці визначають наявність чи відсутність відношення інцидентності між вершинами та ребрами.

Використовуеться рідко із-за відсутності алгоритмів обробки працюючіх з такою структурою.

в) Представлення за допомогою списків суміжностей є головною альтернотивою представленю за допомогою матриць. Список суміжностей для вершини v є списком кінцевих дуг, що виходять із циєї v вершини орграфу, або просто списком всіх суміжних із v вершиннеоріентованого графу.

Наведемо приклад спискового представлення графів, що мали наведення вище матричне представлення для неорієнтовоного G1:

x1 : x2, x4; x3 : x2, x4; x4 : x1, x2, x3, x5, x6;

x2 : x1, x2, x4; x5 : x4, x6; x6 : x5, x6;

представлення та орієнтованого графа G2:

x1 : x2, x3; x2 : nil; x3 : x2, x4, x5; x4 : x5; x5 : x6; x6 : x4;

г) Представлення за допомогою списка дуг використовують для збереження різної інформації про дуги. При цьому способі кожній дужзі придають трійку чисел(u, x, y) ,де u=(x ,y), x-початок, y-кінец дуги. Вагу дуги можливо представити як четверте число до цієї трійки.

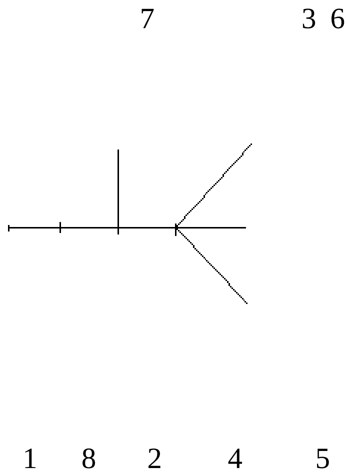
Код Харари визначаемо за допомогою матриці A(G)- матриці суміжностей шляхом послідовного запису рядків із тих елементів, що розміщені над головною діагональю, один за одним. Таким чином матимемо двійкове число, величина якого залежитеме від нумерації вершин. Найбільше ізціх чисел буде буде кодом Хірарі данного графу G. Нумерція вершин, що відповідае коду Хірарі зветься каноничною.

Код Прюфера **використовуеться для представлення дерев. Нехай T-дерево із множиною вершин { v1, v2,…vn }, де номер вершини viдорівнюе i. Припишемо дереву T послідовніст (a1,a2,…an-2)побудовану за наступним правиломЖ**

1) i=1;
2) в послідовності 1,2,…n шляхом перегляду зліва на право шукаемо номер першої висячої вершини. Нехай це bi .
3) Щукаемо вершину що суміжна із bi . Нехай це ai.
4) В послідовності із пункту 2) викреслюемо bi .

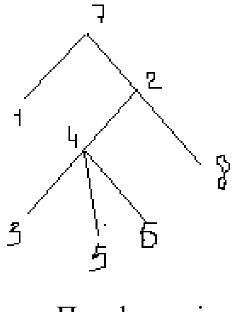
5) В дереві Твидаляемо вершину bi.
6) i = i+1;
7) якщо i< n-1, то переходимо до 2), інакше видаемо { ai,…an-1 }

Це й буде код Прюфера. Наприклад для дерева:



код Прюфера матемо вигляд (8,4,4,4,2,2).

У випадку ордерева побудова коду Прюфера виконуеться аналогічно.Необхідно тільки на останьому місці писати кореневу вершину та при декодуванні коду недописувати цю вершину. Так для ордерева:



Матимо кодПрюфера рівним(7,4,4,4,2,2,7).

Код Прюфера е оптімальним з точки зору економії пам’’яті та доведений теоремі Келі :

n-2

числом помічених n-вершиндерев=n .

№2-2 **Глобальний аналіз графів.**

Означае виділення структури графа та визначення характеристик виделенної структури для розв'язку задачі. Цей аналіз полягае в збіранні інформаціїпро побудову графа шляхом обходу вершин та дуг(ребер) графа. Інформація отримана таким шляхом оформлюеться у виглядіпідходящої нумерації вершин графа.

1 Нумеція ,що виявляе логічну структуру графа.

1.1. Нумерацією F будемо називати приписування вершинам графа G різних чисел (номерів) з множини натуральних чисел N , то F : V(G) <-> N . Звеликого касу нумерації найбільш важливишими е нумерація побудована на пошуку вглибину(базисна нумерація), пряма нумерація та еранжировка. Іноді ці нумерації звуть лінійними.
Пошук в глибину це обхід вершин графа за наступних правил:

1) Знаходячісь у вершині x треба рухатися в любю іншу, раніше не пройдену, якщо така знайдеться, одночасно запам'ятовуючі дугу по якій вперше попали до вершини ;
2) Якщо із вершини x неможливо потрапити до раніше пройденої вершини або такої взагалі немае, то повертаемося до вершини zіз якої вперше попали до x та продовжумо пошук в глубену із вершени z.

При виконанні обходу графа поцім правилам ми намагаемося проникнути в глибь графа наскількі це можливо , потім відступаемо на крок назад і знову намагаемося пройти в перед. При пошуку в глиб орграфа можливопопасти в вершину y з вершини x тількі завдякі наявності дуги (x,y), то ми повинні рухатися вперед тількі в напрямку орієнтації дуг, а повертатися в протележному напрямку. Внеорентованому графі таких обмежень немае. Будемо називати М-нумерацію вершин графа ту нумерацію що відповідае порядку їх обходу при пошуку в глубину. Шлях м=(g=x1,x2,…,xn=p) називатиме М-шляхом, якщо для кожної i,i=1,(1)n, виконуеться умова:

M(xi)<M(xi+1), то номер xi меньше номера xi+1; Вершина p зветься М-досяжною із вершини g, якщо існуе М-шлях із g в p

1.2 Алгоритм пошуку в глубину та побудову М-нумерації в орієнтованому графі мае наступний вигляд:

Вхід : Граф G=(V,E) заданий списками суміжностей A(v) , де v – вершина з множини V, A(v) її список суміжних вершин.

Вихід: М-нумерація вершин і розбиття множин E на чотири класи: деревяних дуг T , прямих дуг F , обернених дугB та поперечних дуг C.

початок

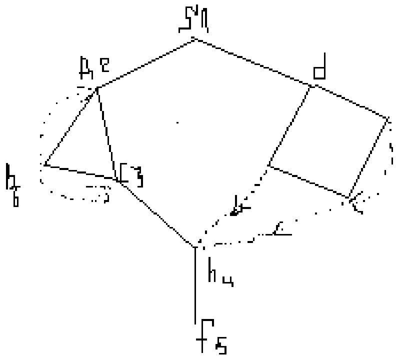
прцедура ПОШУК (V,N);

початок;

1. присвоїти виршині v номер M(v)=N;
2. N:=N+1;
3. Для w E A(v) // вершина w вибрана із списку A(v)
   Виконати цікл:

4. якщо w не мае М-номера то
5.   початок;
6.   добавити дуги (v,w) в T ;
7.   Пошук (w, N) ;
8.   Кінец;
9.   інакше;
10.   початок
11. ___   якщо М-номер вершини w більше М-номера вершини v то доьавити (v,w) до F,
12. ______інакше якшо існуе M –шлях із w в v то
13.   добавити (v ,w) до B;
14.   інакше   добавити (v ,w) до C;
15.   кінець // якщо з g)
16.   кінець // циклу
17.   T:=0; F:=0; B:=0; C:=0; N:=1;
18. для v E V виконати в циклі
19.   помітить вершину v як немаючу номера
20. поки існуе вершина v без М-номера виконати
21.   ПОШУК (v, N);
22. кінец циклу поки. // кінец алгоритму.

Приклад М- нумерації побудованої пошуком вглиб:



Де жирними дугами помічено деревьяні дуги з множини T, тонкими дугами помічено прямі дуги з множиниF, штриховими лініями поміченно обернені дуги із множини B, а пунктиром поміченно поперечні дуги із множини C.

Використання стеку спрощуе реалізацію алгоритма. Присвоення M- номера відбуваеться в той момент коли вершина вводиться в стек; видалення вершини відбуваеться в той момент коли виявляються пройденими всі дуги,що виходять із данної вершини.

N-нумераціею вершин графа, що мае n-вершин, зветься присвоення перший викинутий зі стеку вершині номера n, а другій викінутй вершині присваемое номер n-1.

Пошук в глибину у неоріентованому графі відризняеться від пошуку в глибину в орграфі тим, що всі ребра розбаваються на три класи:

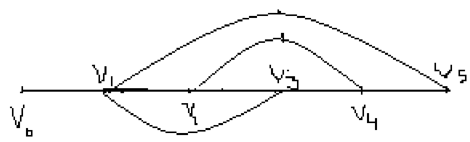
Клас деревьяних(остовних) ребер, клас ребер дотику та клас прямих ребер, де клас ребер дотику відповідае класамобернених та поперечних дуг в орграфі т.то B+C. Цім вичерпууться зміни в наведеному алгоритмі.

Пошук в глибинуу неорентованому графі перетворюе вхідний графG на орграф G' , шляхом наведення на кожному ребрі оріентації у напрямку проходження.

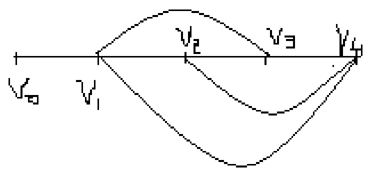
1.3.

Аранжировкою або A-нумераціею будемо називати таку нумерацію коли кожен простий шлях із входом A однохідного графа зветься A-шляхом. Граф зветься аранжируемим якщо він допускае аранжировку.

Приклад аранжуруемого наведемо як граф G1:



А граф G2 е неаранжуемим:



Не факт неаранжуемості вказуе наявність двох шляхів із входу s=v0 в вершину v2, один із яких містить вершину v3, а другий ні.

2-4

Побудова аранжировки для аціклічного графа здійснюеться за наступним алгоритмом:

<u>Вхід:</u>   Граф G=(V,E) заданий стекамі суміжностей A(v) , де V0 множина вхідних вершин.

<u>Вихід:</u> Аранжировка (А-нумерація) вершин графа.

1. початок
2. процедура Аранжіровка (v, N);
3. початок
4. присвоїти вершині v номер N ;
5. N:=N+1;
6. Для w E A(v) ЦІКЛ
7. Виконати Аранжировка (w, V), якщо всі попередні вершини w маютьА-номери
8. Кінец
9. N=1
10. Для v E V0 цікл
11. Помітити вершину v як не маючу А-номера; кінець циклу
12. Покі існуе вершина v E V0 без А- номера виконати цикл АРАНЖИРОВКА(v ,N);
13. Кінец циклу поки
14. кінец алгоритму

1.4.

Пряма нумерація виконуеться на базі М-нумерації та полягае у визначенні формальних циклів графа, де під формальним циклом розуміють граф породженний вершинами простого шляху F(i,. .k) та оберненої дуги (Vk, Vi) вважаемо що від ш-тої вершини до k-тої зростають (i < j < k).

№2 -4 Логічний аналіз графів. Лінійні компоненти.

2.1. Матриця досяжності та транзетивне замикання.

У випадку , коли необхідно встановити ліше факт наявності шляхів між вершинами, виникае задача про побудову транзитивного замикання орграфа. Транзетивним замиканням G* графа Gзветься орграф із тіеюже множиною вершин що й G, але в ньому присутня дуга (xi xj) тоді і тількі тоді , коли вершина xj досяжна із xi ,.т. то існуе шлях (xi- xy) . Матрицею суміжностей транзетивного замикання G* орграфа G служить матриця досяжності R(G) графа G , т. то квадратна матриця порядка n , n= /V(G)/ , із елементами rij=1,якщо вершина xj досяжна із xi , также rij =0.

Найпростішим алгоритмом побудови матриці R(G) мае вигляд:

Вхід: Матриця A(G) суміжності орграфа G , що мае рядки A1, A2, .. An.

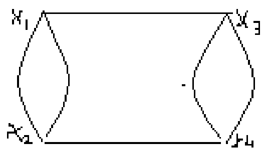
Вихід: Матриця досяжності R(G) ,що мае рядки R1,R2, ..Rn.

Початок алгоритма:

1. для і від 1 до nкрок 1 виконати цікл
2. побудувати множину J <-> { j }, таких індексів ,що aij=1;
3. Ri:=Ai; K:=0;
4. Поки J<-> 0 (J непуста множина) виконати цикл
5. Вибрати j E J;
6. Ri:=Ri+Aj, де + -- логічна операція ’ и ’, тобто обеднаня матриць одного порядку , тобто 1 або в одній або в другий матриці дае A в результативній матриці.
7. J:=J\{ y };
8. K:=K U { j };
9. Зформувати множену Ji – індексів K таких , що ajk=1;
10. J:=J U (Jj \K);
11. Кінец цикла;
12. Виводимо Ri;
13. Кінец цикла ;

Кінец алгоритму.

Наприклад, для графа G та його матриці A(G) :



0110

1000

: A(G) = 0001

0010

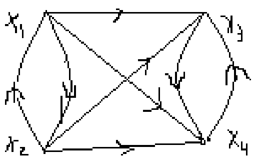
маемо матрицу R(G) та G* -транзетивне замикання:

1111

1111

R(G)= 0011

0011



Більш ефективним е алгоритм Уоршела (Warahall) , якій знаходить матрицу R шляхом обчислення послідовності квадратних матриць

0 1 n

B , B , … B порядка n за наступними правилами:

0 e e

1. B : = A(G) ; // B = {B ij};
2. Для L =1 до n кроком1 виконати цикл
L L-1 L-1

3. B ij = B ij +|B ij & Bij | ; // + логічне додавання діеть множину ребер із числа присутніх або в одному або в другому
4. Кінець циклу

n

5. B=B ;

Кінець алгоритму.

e e

Елементам B ij надати той зміст , що B ij=1 тоді ітільки тоді коли ершина xi та xj звязані шляхом , що проходить через вершини xi, x…xj .

0

Тоді початковий крок B = A(G) означатиме шляхи без прміжних вершин, а останній крок

n

R:=B означатиме наявність шляхів через любі проміжні вершини. Цикл виконувате провірку факту, що xi та xj звязанні шляхом , проміжні вершини якого належатимуть множині { xi,…xj } та задовільняють випадкам:

1) існує шлях від xi до xj , що проходить через (xi,…x L-1 };
2) існує шлях від xi до x L та від x L до xj , всі проміжні вершини якого належать множині (xi,…x L-1 }.

4-2

2.2.

№ 2-3 <u>Відшукання бікомпонент орграфа.</u>

Бікомпонентою називають найбільший (за включенням) підграф в якому люба пара вершин досяжна із кожної вершини іншої пари вершин. Ефективний алгоритм знаходження біокомпонент на пошуку “ в глибину ” мае наступний вигляд:

<u>Вхід</u>: Граф_G =(V,E), заданий списками суміжностей A(v).

<u>Вихід</u>: Список B з біокомпонент.

Початок алгоритму

1. Процедура БІОКОМП (End (TT)) ;
2. Початок процедури
3. V : = End (TT);

4. Для w E A(v) виконати цикл
5. Якщо A(v) недорівнюють 0 ,то
6-8. якщо S(w) =0 то 1) TT : = w; S(w) =1

2) БІКОМП (w);

9.
10. інакше // w розміщенно в стеку TT
11-12. стягнути хвіст стеку TT починаючі із вершини w до вершини w' , т.то видаляти, починаючі із вешини w хвіст TT, запамятовувати видаленні вершини та замість w взяти w' для якої список суміжностей е списком отримань шляхом обеднання списків суміжностей зтягнутих вершин.
13. БІКОМП(w') ;

14. інакше 1) видаити v із стеку TT та занести в B, т.то в список біокомпонент вершину v;
2) БІКОМП (End (TT));

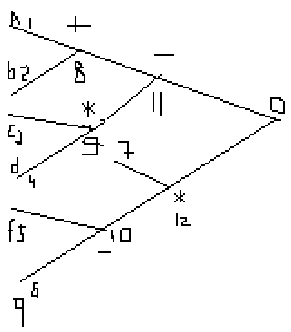
15. кінец циклу;
16. повертаемо B;
17. кІнец процедури БІКОМП ( );
18. кінец алгоритму.

7-1

7.2.

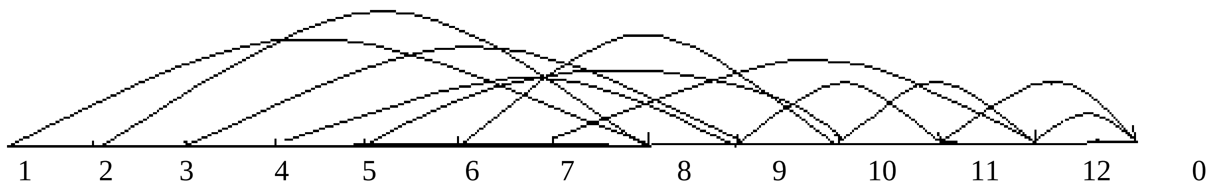
Мінімізація памяті при обчисленні арифметичних виразів.

Арифметичні вирази можуть відображатися як ордерева, в кожну вершину яких входить не більш двох дуг. Початкові данні відповідають висячим вершинам дерева, а проміжні результати обчислень. Кожна внутрішна вершина відображае бінарну операцію над оргументами, представленими вершинами- попередниками. Порядок в якому треба виберати аргументи несуттевий у простих моделях.Наприклад процес обчислення арифметичного виразу ((a +b)-c * d)



/ (e * (f - g)) зображенна наступним чином:

Вибір можливого порядку операцій означае топологічно впорядкувати дерево ,т.то розтошувати його вершини у ‘ цілочисельних ’ точках числової прямої в такій послідовності де вершина v передуе вершини v’ якщо існуе дуга із v’ в v, т.то дуга (v’, v) . Ця умова еквівалентна вимозі , щоб проміжний результат обчислювався раніше ніж використовувався. Приклад топологічного впорядкування дерева , для наведеного вище виразу мае вигляд:



Відзначимо, що величини які не використовуються в якості операндов- аргументів операції , що виконуеться у поточний момент часу , повинні зберігатися в памяті.

На малюнку топологічного впорядкування дерева тиким велечинам відповідають дуги , які проходять над вершиною що позначае поточну операцію.

Таким чином виникае оптимізаційна задача:

Мінімізувти кількістькомірок памяті для організації обчислення арифметичних виразів.

Ця задача зводиться до побудови топологічних впорядкувань ордерева із мінімальною ширеною.

Генерація оптимального коду для арифметичних виразів.

Нехай маемо машину із необмеженою памятю та Т-регістрами та допустимими е команди наступних типів:

1) переселання: память → регістр;
2) переселання: регістр → память;
3) Операція : (регістр, память) → регістр;
4) Операція : (регістр, регістр) → регістр.

Зауважимо на те , що операція: (регістр, память) → регістр; е забороненою. Під ціною обчислень будемо розуміти кількість операторів (програмних кроків) потрібних для повного процесу обчіслень. Для того, щоб визначити наймеьшу кількість потребних регістрів, а також оптимальну послідовність операцій будемо використовувати розмітку вершин, враховуючі некомутативні оерації. Розмітку виконаемо наступним чином:

1) якщо вершина v висяча та е лівим потоком деякої вершини, то покладемо L(v)=1, якщо вона е правим потоком , то вважатимемо L(v) = 0/

2) якщо вершина v мае потоків із мітками L1, L2; то при L1#L2 покладемо , що L(v) =max (L1,L2) ,а при L1=L2=L вважатимемо , що L(v) =L+1.

Алгоритм розвязку мае наступний вигляд:

Початок алгоритму

1. процедура T (v);
2. початок процедури
3. якщо f(v) то
4-5. якщо v-вісяча вершина ,то заслати значення вершини v у доступний регістр Bm із найменьшим номером; // в цьому випаку вешина v е лівим потоком свого предка //
6. інакше T(&(v)) ;
7-8. інакше якщо min (L(&(v)) , L (p(v))) => N то T (p(v));

// користуемося позначенням &(v) для лівого left(v), а через P(v) позначмо right(v) – правого потомка для v.

9. інакше якщо L(&(v))# L (P(v)) то
10. початок
11. w:= потомок вершини v із найбільшою міткою.
12. T(w);
13. кінец;
14. інакше T(P(w));
15. якщо (L(v)=1) ^(v-не е висячою вершиною) то
16. початок
17. обчислити вершину v , взявши значення лівого потомка із регістру Bm , а значення правого потомка із памяті
18. заслати значення вершини v до регістру Bm;
19. кінец;
20. інакше початок
21. T(&(v));
22. виконати операцію , що відповідае вершині v , беручі значення лівого потомка з регістру Bm , а значення правого потомка з регістру Bm+1;
// на цьому кроці доступними е регістри Bm , Bm+1;

23. якщо (L(v) => N)^(v- правий птомок свого предка) то

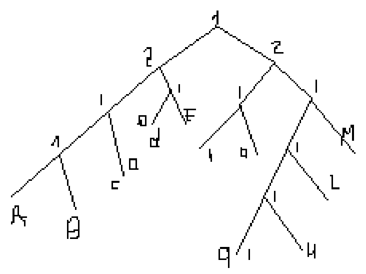
24. початок

25. значення вершини v заслати в память;
26. звільнити регистри Bm , Bm+1;
27. кінець;
28. кінець // якщо з номером 3
29. кінец процедури T(v)
30. v = корень дерева T.
кінець

Розглянемо приклад роботи цього алгоритму над виразом

((ab-c) /(d+c))/(g+i)/(j+k)*L-m)).

Дерево для цього виразу мае вигляд:



Для N=2 алгоритм генерує такий код:

1. f→ B1;
2. B1+k → B1;
3. B1*L → B1;
4. B1-m→ B1;
5. G → B2;
6. B2+I→ B2;
7. B1/B2→ (память);
8. D→B1;
9. B1+e→ B1;
10. A→ B2;
11. B2*B→B2;
12. B2-c→ B2;
13. B1/B2→B1;
14. B1/(память)→ B1;

Практична робота №3

ТЕМА: Лінійний синтез скінчених графів

МЕТА: Отримати навички лінійного синтезу дискретніх об'єктів та аналізу наслідування властивостей.

Завдання. Виконати синтез по всім різним простим ланцюгам, як вручну, так і за допомогою програмного засобу, двох наступних графів:

1) заданного графа із додатку 1 та порядковим номером тотожнім номеру Вашого прізвища в журнальному списку;

2) графа $K_{2,3}$ для першої групи, графа K_4 для другої групи

та проаналізувати число досяжності множини вершин кожного синтезованого графа.

Приклад:

1) Рассмотрим граф G (см. рис. 3.) иллюстрирующий

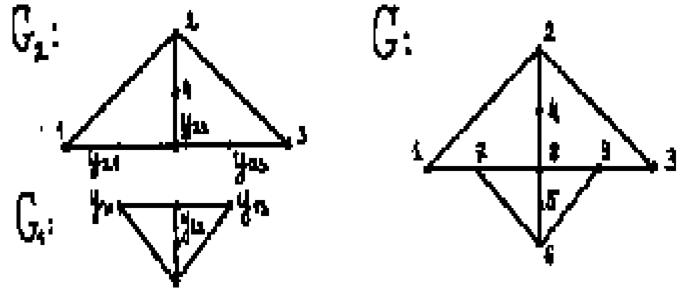


Рис. 3.

Граф G является φ - образом графа $\sum_{i=1}^{2} G_i$, где $G_2 \cong K_{2,3}$ $G_1 \cong K_{2,.3}$ а φ - преобразование задано следующим образом:

$$\varphi(\sum_{i=1}^{2} G_i, \sum_{j=1}^{2}(y_{1j} + y_{2j})) = (G, \{y_j\}_{j=1}^{3})$$

где $y_j = 6 + j, j = 1,2$,

а) $G_i \cong K_{2,3}, i = 1,2;$

б) $G_j(\{y_{ij}\}_{j=1}^{3})$ - простая цепь длинны 2 графа G_i

y_{21}, y_{23} - внутренние точки ребер, $i = 1,2$

в) $G(\{y_i\}_{j=1}^{3})$ - простая цепь длинны 2 графа G.

2) Рассмотрим граф G (рис.4.) иллюстрирующий утверждение 3) теоремы 1.2. в том случае когда $G_0(\{Z_{0j}\}_{j=1}^{n})$ - простой цикл, не являющийся границей внешней грани графа $f(G_0)$, где вложение f реализует $t_G(G^0)$.

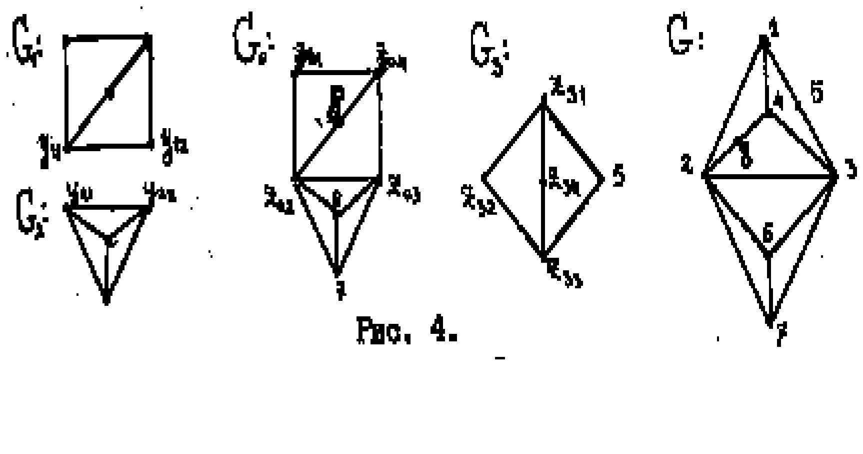


Рис. 4.

а) φ - преобразование графа $\sum_{i=1}^{2} G_1$ в граф G_0 следующим образом:

$\varphi(\sum_{i=1}^{2} G_i, \sum_{j=1}^{2} y_{1j} + y_{2j}) = (G_0, \sum_{i=2}^{3} Z_{0i})$,

где $G_j(\{y_{ji}\}_{i=1}^{2})$ - ребро графа G_j, $j = 1,2$,

$G_0(\{Z_{0i}\}) \in G^1$;

б) $G_3 \approx K_{2,3}$;

в) $G_0(\{Z_{0i}\}_{j=1}^{4})$, $G_3(\{Z_{3j}\}_{j=1}^{4})$, $G(\{j\}_{j=1}^{4})$ - простые циклы длинны 4 графов G_0, G_3, G - соответственно.

Практична робота №4

ТЕМА: Нелінійний синтез скінчених графів

МЕТА: Отримати навички нелінійного синтезу дискретніх об'єктів та аналізу наслідування властивостей.

Завдання. Виконати синтез по всім різним простим циклам, як вручну, так і за допомогою программного засобу, двох наступних графів:

1) заданного графа із додатку 1 та порядковим номером тотожнім номеру Вашого прізвища в журнальному списку;

2) графа $K_{2,3}$ для першої групи, графа K_4 для другої групи

та проаналізувати число досяжності множини вершин кожного синтезованого графа.

Приклад. Для иллюстрации в том случае, когда $G_0(\{Z_{0i}\}_{j=1}^{4})$ - цикл с диагональю, являющийся внешней гранью графа $f \mid G_0(G_0)$, где f - вложение реализующее $t_G(G^0)$:

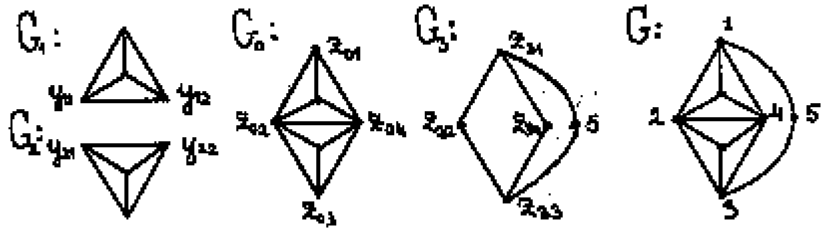


Рис. 5.

Практична робота №5-6

ТЕМА: Алгоритми на скінчених графах

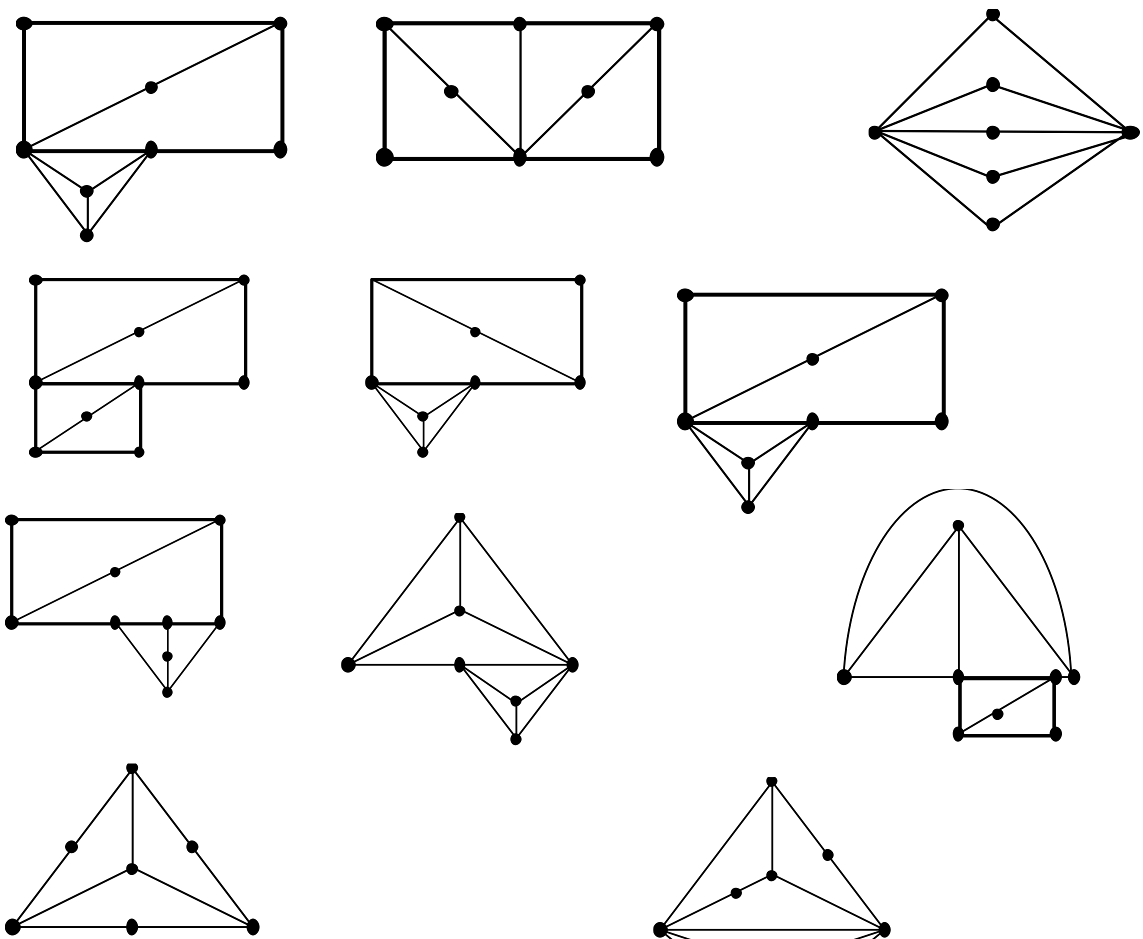
МЕТА: Отримати навички

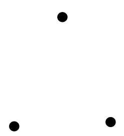
Завдання: 1) Виконати побудову остовного дерева графа із додатку 1 номер якого співпадає із номером вашого прізвища в списку групи, як вручну, так і за допомогою программного засобу методом в глибину графа;

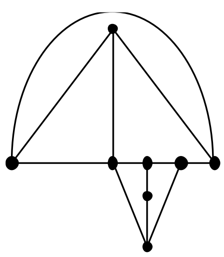
2) Виконати побудову множин фундаментальних циклів та множин простих циклів графа із додатку 1 номер якого співпадає із номером вашого прізвища в списку групи, як вручну, так і за допомогою программного засобу методом в глибину графа.

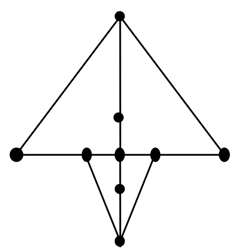
Додаток 1

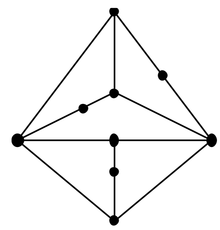
Графи 3-минимальні із номерами №1-№18.

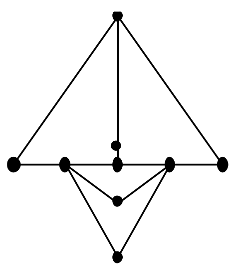


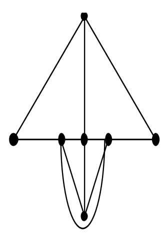












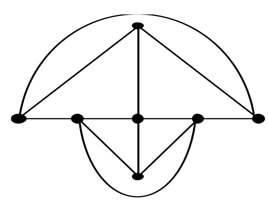


Рис.1. Графи 3-минимальні із номерами №1-№18.