Примерни решения на задачите от изпита по Дискретни Структури, КН 2 поток, 11 февруари 2022 г.

Задача 1. Дефинираме "композиция на функции" по следния начин. Нека A, B и C са произволни множества. Нека $\phi : A \to B$ и $\psi : B \to C$. Функцията $\psi \circ \phi : A \to C$, дефинирана така:

$$\forall a \in A : (\psi \circ \phi)(a) = \psi(\phi(a))$$

наричаме *композицията на* ψ *и* ϕ. Внимание: в израза $\boxed{\psi \circ \phi : A \to C}$, $\boxed{\psi \circ \phi}$ е името на функцията-композиция.

Нека X и Y са произволни множества. Нека $f : X \to Y$, $g : Y \to X$ и $i : Y \to Y$. Нека $\forall y \in Y : i(y) = y$. Нека $f \circ g = i$. Докажете или опровергайте, че f е сюрекция.

Решение. Твърдението е вярно. Да разгледаме произволен елемент $y \in Y$. Нека $x = g(y)$. Щом f е тотална функция, $f(x)$ е дефиниран. Но

$$\begin{aligned} f(x) &= f(g(y)) \quad // \text{ по определението на } x \\ &= (f \circ g)(y) \quad // \text{ по определението на композиция} \\ &= i(y) \quad // \text{ по условие} \\ &= y \end{aligned}$$

Показахме, че за всяко $y \in Y$ съществува $x \in X$, а именно $x = g(y)$, такъв че $f(x) = y$. Но това е определението на сюрекция. Показахме, че f е сюрекция.

Задача 2. Дефинираме "композиция на цяло положително число" по следния начин. За всяко цяло положително n, *композиция на* n е всеки вектор от цели положителни числа, чиято сума е n. Примерно, $(1, 2, 1, 3)$ е композиция на 7, а $(1, 1, 3, 2)$ е друга композиция на 7. След това дефинираме, че k-*композиция на* n е всяка композиция на n с точно k елемента. Примерно, $(1, 2, 1, 3)$ е 4-композиция на 7, а $(1, 1, 3, 2)$ е друга 4-композиция на 7. $(1, 1, 1, 1, 1, 1, 1)$ е единствената 7-композиция на 7, а (7) е единствената 1-композиция на 7.

Колко k-композиции на n има?

Колко композиции на n има?

Решение. Нека $C(k, n)$ е множеството от k-композициите на n. Нека $V(k, n)$ множеството от векторите, състоящи се от $k-1$ елемента от един вид, да ги бележим с "$|$", и n елемента от друг вид, да ги бележим с "$*$", които вектори не може да започват с $|$, не може да завършват с $|$ и не може да има съседни елементи $|$. Съществува очевидна биекция от $C(k, n)$ във $V(k, n)$. Примерно,

- на композицията $(1, 2, 1, 3)$ от $C(4, 7)$ съответства векторът $*|**|*|***$ от $V(4, 7)$,
- на композицията $(1, 1, 3, 2)$ от $C(4, 7)$ съответства векторът $*|*|***|**$ от $V(4, 7)$,
- на композицията $(1, 1, 1, 1, 1, 1, 1)$ от $C(4, 7)$ съответства векторът $*|*|*|*|*|*|*$ от $V(4, 7)$,
- на композицията (7) от $C(4, 7)$ съответства векторът $*******$ от $V(4, 7)$.

Елементите $*$ отговарят на единиците-елементарни събираеми, от които се състои n, а вертикалните разделители показват как са групирани тези единици в k ненулеви събираеми. Редът на събираемите има значение и наредбата на групите $* \cdots *$ във векторите на $V(k, n)$ съответства точно на реда на събираемите във векторите на $C(k, n)$. Ясно е защо не може да има $|$ в началото или в края: това би съответствало на събираемо нула, което не е разрешено. Съседни елементи $|$ не са разрешени по същата причина.

Търсим $|V(k, n)|$. Съобразяваме, че вертикалните разделители са $k - 1$ на брой и има $n - 1$ позиции, на които можем да ги слагаме. Всеки избор на $k - 1$ от общо $n - 1$ позиции определя еднозначно едно от слаганията на вертикалните разделители. Заключаваме, че $|V(k, n)| = \binom{n-1}{k-1}$. Но тогава $|C(k, n)| = \binom{n-1}{k-1}$ по принципа на биекцията.

Броят на всички композиции на n се получава със сумиране по броя на k-композициите на n, по всички k от 1 до n. А именно,

$$\binom{n-1}{0} + \binom{n-1}{1} + \binom{n-1}{2} + \cdots + \binom{n-1}{n-1}$$

Съгласно изучаваното на лекции, тази сума е 2^{n-1}.

Задача 3. Докажете с комбинаторни разсъждения, че за всеки цели положителни m и n е изпълнено

$$\sum_{k=0}^{2n+2} (-1)^k \binom{n}{k} (n-k)^m = \begin{cases} 0, & \text{ако } m < n \\ \dfrac{1}{\sum\limits_{k=0}^{n} \binom{n}{k}^2} \cdot \dfrac{(2n)!}{n!}, & \text{ако } m = n \end{cases}$$

Решение. От лекции знаем, че биномният коефициент е нула, ако долният индекс е по-голям от горния индекс. Тогава биномните коефициенти $\binom{n}{n+1}$, $\binom{n}{n+2}$, …, $\binom{n}{2n+2}$ са нули, така че може да опростим лявата страна по следния начин:

$$\sum_{k=0}^{n} (-1)^k \binom{n}{k} (n-k)^m = \begin{cases} 0, & \text{ако } m < n \\ \dfrac{1}{\sum\limits_{k=0}^{n} \binom{n}{k}^2} \cdot \dfrac{(2n)!}{n!}, & \text{ако } m = n \end{cases}$$

Това е тъждеството, което се иска да докажем.

От лекции знаем, че $\sum\limits_{k=0}^{n} \binom{n}{k}^2 = \binom{2n}{n}$. Тогава

$$\frac{1}{\sum\limits_{k=0}^{n} \binom{n}{k}^2} \cdot \frac{(2n)!}{n!} = \frac{1}{\binom{2n}{n}} \cdot \frac{(2n)!}{n!} = \frac{1}{\frac{(2n)!}{n! \cdot n!}} \cdot \frac{(2n)!}{n!} = n!$$

Предвид това, тъждеството, което се иска да докажем, е

$$\sum_{k=0}^{n} (-1)^k \binom{n}{k} (n-k)^m = \begin{cases} 0, & \text{ако } m < n \\ n!, & \text{ако } m = n \end{cases}$$

Но лявата страна е броят на сюрекциите с m-елементен домейн и n-елементен кодомейн, съгласно изучаваното на лекции. Ако мощността на домейна е по-малка от мощността на кодомейне, сюрекции няма, тоест броят им е нула. Ако мощностите на домейна и кодомейна са равни, всяка сюрекция е и инекция, следователно и биекция, а знаем, че биекциите са $n!$ на брой, където n е мощността на домейна и кодомейна.

Задача 4. Нека $A = \{0, 1, 2, 3, 4, 5\}$ и $n \in \mathbb{N}$.

1 т.
- Колко са векторите с дължина n, чиито елементи са от A?
- Колко са векторите с дължина n, чиито елементи са от A и в които няма съседни числа, които не се делят на 3?

12 т.
  - Съставете подходящо рекурентно уравнение.

7 т.
  - Решете уравнението.

Решение. Тъй като $|A| = 6$, векторите с дължина n над A са 6^n.

Да видим колко от тях нямат съседни числа, които не се делят на 3. От елементите на A, на 3 се делят 0 и 3, а 1, 2, 4 и 5 не се делят на 3. Иска се да бъдат преброени векторите, които нямат съседни елементи от $\{1, 2, 4, 5\}$.

Нека S_n е търсеният брой. Иска се да съставим рекурентно уравнение за S_n. Очевидно $S_0 = 1$ и $S_1 = 6$, защото при дължини 0 и 1 няма "нарушители". За $n \geq 2$ разсъждаваме така.

- Ако първото число е 0 или 3, останалата част от вектора не трябва да съдържа съседни елементи от $\{1, 2, 4, 5\}$. Но останалата част е с дължина $n - 1$. Заключаваме, че рекурентното уравнение има събираемо $2S_{n-1}$ вдясно.

- Ако първото число е от $\{1, 2, 4, 5\}$, то второто задължително трябва да е 0 или 3, а останалата част от вектора не трябва да съдържа съседни елементи от $\{1, 2, 4, 5\}$. Но останалата част е с дължина $n-2$. Заключаваме, че рекурентното уравнение има събираемо $8S_{n-2}$ вдясно. Коефициентът 8 е заради това, че всяко начало от $\{1, 2, 4, 5\}$ може да се комбинира с втори елемент от $\{0, 3\}$.

- Други събираеми вдясно няма, защото горните два случая са изчерпателни.

И така, рекурентното уравнение, заедно с началните условия, е

$$S_n = \begin{cases} 1, & \text{ако } n = 0 \\ 6, & \text{ако } n = 1 \\ 2S_{n-1} + 8S_{n-2}, & \text{ако } n \geq 2 \end{cases}$$

Това е линейно хомогенно рекурентно уравнение с константни коефициенти и крайна история. Знаем как се решават такива уравнения.

Характеристичното уравнение е $x^2 - 2x - 8 = 0$. Корените му са 4 и -2, откъдето общото решение е

$$S_n = A4^n + B(-2)^n$$

за някакви константи A и B. От началните условия имаме

$$\begin{aligned} 1 &= A + B \qquad // \text{ за } n = 0 \\ 6 &= 4A - 2B \qquad // \text{ за } n = 1 \end{aligned}$$

Намираме $A = \frac{4}{3}$ и $B = -\frac{1}{3}$. Оттук

$$S_n = \frac{4}{3}4^n - \frac{1}{3}(-2)^n = \frac{1}{3}\left(4^{n+1} - (-2)^n\right)$$

Задача 5. Докажете, че във всеки планарен граф има връх от степен, не по-голяма от 5. Има лесно доказателство с допускане на противното.

- Първо напишете ясно и прецизно противното твърдение.

- Какво следва за броя на ребрата от противното твърдение? Открийте противоречие между това и нещо, изучавано на лекции.

Докажете по индукция по броя на върховете, че $\chi(G) \leq 6$ за всеки планарен граф G.

Решение. Противното твърдение е:

Съществува планарен граф, в който всеки връх е от степен поне 6.

Да допуснем, че то е вярно.

От лекции знаем, че за всеки граф $G = (V, E)$ е в сила $\sum_{v \in V} d(v) = 2m$. Ако $d(v) \geq 6$ за всеки връх v, то $\sum_{v \in V} d(v) \geq 6n$. Тогава $2m \geq 6n$, тоест $m \geq 3n$. От лекции обаче знаем, че за всеки планарен граф е изпълнено $m \leq 3n - 6$. Полученото противоречие доказва, че няма планарен граф, в който всеки връх е от степен поне 6. С други думи, във всеки планарен граф има връх от степен ≤ 5.

Ще използваме този факт, за да докажем по индукция по броя на върховете, че $\chi(G) \leq 6$ за всеки планарен граф G

Базата е $n = 1$. Твърдението очевидно е вярно.

Да допуснем, че всеки планарен граф с не повече от k върха може да бъде оцветен с най-много 6 цвята. Разглеждаме произволен планарен граф $G = (V, E)$ с $k + 1$ върха. Съгласно доказаното преди малко, в G има връх u от степен ≤ 5. Нека $N(u)$ е множеството от върхове, съседни на u. Изтриваме u от G и получаваме граф $G' = (V', E')$ с k върха, където $V' = V \setminus \{u\}$. Но G' е оцветим с най-много 6 цвята съгласно индуктивното предположение. Нека $f : V' \to C$ е оцветяване на G' с най-много 6 цвята; C е множеството от цветовете и $|C| \leq 6$.

Върховете от $N(u)$ присъстват в G', понеже изтриването на u не ги "засяга". Но $|N(u)| \leq 5$. Нека $D = \{f(z) \mid z \in N(u)\}$. На прост български, това са цветовете, които се ползват за оцветяването на върховете от $N(u)$. Очевидно $|D| \leq 5$. Тогава има поне един цвят $c \in C$, който не е в D; тоест, c не е цвят на никой връх от $N(u)$.

Сега добавяме обратно u към G' заедно с ребрата между него и върховете от $N(u)$, получавайки оригиналния граф G. Оцветяваме u в цвят c и забелязваме, че при това получаваме коректно оцветяване на G, понеже всеки от $N(u)$ е в някакъв друг цвят. Нещо повече, това е оцветяване на G с не повече от 6 цвята.

Доказахме, че $\chi(G) \leq 6$.

Задача 6. Припомнете си алгоритъма на Дийкстра за намиране на най-къси пътища в тегловен граф с положителни тегла от връх s до всички останали върхове. Може ли да получим от него алгоритъм за **най-дълги** пътища в тегловен граф с положителни тегла от връх s до всички останали върхове, ако обърнем неравенствата и заменим ∞ с $-\infty$ и заменим минимум с максимум? Става дума за следните промени.

1. Инициализираме стойностите на върховете не с ∞, а с $-\infty$.
2. Проверката за край на алгоритъма става "ако извън множеството S няма връх със стойност, по-голяма от $-\infty$, край". В алгоритъма на Дийкстра тази проверка е "ако извън множеството S няма връх със стойност, по-малка от ∞, край".
3. На всяка итерация избираме връх x, който да сложим в множеството от върхове S, като върхът с **максимална** стойност, който не е в S. В алгоритъма на Дийкстра избираме връх x, който да сложим в множеството от върхове S, като върхът с минимална стойност, който не е в S.
4. При разглеждането на върховете от списъка на съседство на x обръщаме посоката на неравенството така: за всеки връх y в списъка на съседство на x, ако стойността на y е **по-малка** от стойността на x плюс теглото на реброто (x, y), присвояваме на стойността на y стойността на x плюс теглото на реброто (x, y). В алгоритъма на Дийкстра това присвояване се случва, ако стойността на y е по-голяма от стойността на x плюс теглото на реброто (x, y).

Решение. Не можем да конструираме алгоритъм за най-дълги пътища по този начин. Съвсем прост контрапример е неориентираният граф $G = (\{s, u\}, \{(s, u)\})$ с произволна тегловна функция w. За определеност, нека $w((s, u)) = 1$. Стартовият връх е s.

В началото алгоритъмът дава стойност 0 на s и $-\infty$ на u, а S е $\emptyset$. Алгоритъмът проверява дали има връх със стойност, по-голяма от $-\infty$. Такъв има и това е s. x получава стойност s и s влиза в S. За всички съседи на x, а това е само u, се прави проверката; в случая се проверява дали стойността на u е по-малка от стойността на x плюс 1; тоест, дали $-\infty < 0 + 1$. Това е вярно и стойността на u става 1. После x става u, y става s и се проверява дали стойността на s е по-малка от стойността на u плюс 1; тоест, дали $0 < 1 + 1$. Това е вярно и стойността на s бива "подобрена" на 2.

Което е неправилно. Става дума за прости пътища и най-късият път от s до s има тегло 0. В този контрапример, алгоритъмът променя стойността на връх, който вече е в S, "връщайки" се по неориентирано ребро назад.