Писмен изпит (редовни студенти) по ДАА, 01.07.2013г.

Име: __________________________________, ФН:______, Спец./курс:________

Задача	1	2	3	4	5	6	Общо
получени точки							
максимум точки	20	20	20	20	20	20	120

Забележка: За отлична оценка са достатъчни 100 точки!

Задача 1 Решете следните рекурентни отношения:

а) $T(n) = \sqrt{8}T(\frac{n}{2}) + \frac{n^2}{\sqrt{n}}$ б) $T(n) = 2T(\frac{n}{\sqrt{3}}) + \frac{n}{\lg n}$

в) $T(n) = T(n-1) + \frac{1}{n}(1 + \frac{1}{n})$ г) $T(n) = \sum_{i=0}^{n-1} T(i) + 2^n$

Задача 2 Намерете асимптотичната сложност на всеки от следните два фрагмента от програми като функция на n. Допуснете, че функцията `bnm(n,k)`, използвана в `h(n)`, връща $\binom{n}{k}$ и работи във време $\Theta(1)$.

```
int f(int n) {
 int i, s=0, k=1;
 while (k <= n) {
  for(i=0; i<n/k; i++) {
   s++; }
  k++; }
 return s; }
```

```
int h(int n) {
 int i, k=0, t=0;
 if (n < 1) return 1;
 t += h(n-1) + h(n-2);
 while (k <= n) {
  for(i=0; i < bnm(n,k); i++) {
   t ++; }
  k ++; }
 return t; }
```

Задача 3 Даден е неориентиран граф $G_1(V, E_1)$ с теглова функция w. Минимално покриващо дърво в графа T_1 има сума от теглата на ребрата a_1. След премахване на няколко ребра от G_1 бил получен графът $G_2(V, E_2), E_2 \subset E_1$. T_2 е минимално покриващо дърво в G_2 и има сума от теглата на ребрата a_2.

(a - 12 точки) Докажете неравенството $a_1 \leq a_2$.

(b - 4 точки) Дайте пример на двойка графи G_1 и G_2, за които се достига равенство $a_1 = a_2$.

(c - 4 точки) Дайте пример на двойка графи G_1 и G_2, за които се достига неравенство $a_1 < a_2$.

Задача 4 Ориентиран граф $G(V, E)$ не съдържа цикли. Предложете бърз алгоритъм, който по два зададени върха s и t намира броя на всички пътища от s до t.

Задача 5 Съчетание M в неориентиран граф $G(V, E)$ наричаме подмножество от ребра на G, които не се допират (няма ребра в M, които имат общ връх). Върхово покритие C е подмножество върхове на G, които покриват ребрата на графа (за всяко ребро поне единият му край е елемент на C). Докажете неравенствата:

(a) $|M| \leq |C|$

(b) Ако M е максимално съчетание в G, а C е минимално върхово покритие, то $2|M| \geq |C|$.

Задача 6 Разгледайте следните две полиномиални редукции. За всяка от тях отговорете дали е коректна или не. Аргументирайте отговора си.

Редукция 1: SAT $\propto$ 3SAT Нека Q е пример на SAT, тоест конюнктивна нормална форма над множество булеви променливи. За всяка елементарна дизюнкция q:

- ако тя има точно един литерал y, тоест $q = (y)$, добавяме две нови булеви променливи w_1 и w_2, които няма да използваме никъде другаде, и заменяме q с $(y \vee w_1 \vee w_2)$,
- ако тя има точно два литерала y_1 и y_2, тоест $q = (y_1 \vee y_2)$, добавяме една нова булева променлива z_1, която няма да използваме никъде другаде, и заменяме q с $(y_1 \vee y_2 \vee z_1)$,
- ако тя има точно три литерала, оставяме я без промяна,
- и ако тя има повече от три литерала, тоест $q = (y_1 \vee y_2 \vee \ldots \vee y_k)$ за $k \geq 4$, заменяме q с множеството от дизюнкции

$$\{(y_1 \vee u_1 \vee u_2), (y_2 \vee u_3 \vee u_4), \ldots, (y_k \vee u_{2k-1} \vee u_{2k})\}$$

където $u_1, \ldots, u_{2k}$ са нови булеви променливи, които няма да ползваме никъде другаде.

Редукция 2: 3SAT $\propto$ SAT Нека Q е пример на 3SAT. Редукцията връща Q без модификация.

Решения:

Задача 2 Случаи а) и б) са подходящи за решаване с Мастър теорема.

За а) пресмятаме $k = \log_b a = \lg \sqrt{8} = \lg \sqrt{2^3} = \lg 2^{\frac{3}{2}} = \frac{3}{2}$ и сравняваме $n^k = n^{\frac{3}{2}}$ с $f(n) = \frac{n^2}{\sqrt{n}} = \frac{n^2}{n^{\frac{1}{2}}} = n^{\frac{3}{2}}$. Очевидно $n^k = \Theta(f(n))$, попадаме във втори случай на Мастър теоремата, следователно $T(n) = \Theta(n^{\frac{3}{2}} \lg n)$.

За б) пресмятаме $k = \log_{\sqrt{3}} 2 > 1$ и сравняваме $n^k = n^{\log_{\sqrt{3}} 2}$ с $f(n) = \frac{n}{\lg n}$. Очевидно съществува $\varepsilon > 0$, такова, че $n^{k-\varepsilon} \succ f(n)$. Попадаме в първи случай на Мастър теоремата, следователно $T(n) = \Theta(n^{\log_{\sqrt{3}} 2})$.

Рекурентното отношение в) опростяваме до $T(n) = T(n-1) + \frac{1}{n} + \frac{1}{n^2}$ и решаваме със заместване – след $n-1$ замествания и прегрупиране на членовете получаваме:

$$T(n) = T(0) + (1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{n}) + (1 + \frac{1}{2^2} + \frac{1}{3^2} + \ldots + \frac{1}{n^2})$$

Първата сума е хармоничния ред с асимптотична оценка $\Theta(\lg n)$, а втората е част от сходящ ред (можем да я оценим чрез интегриране), следователно $T(n) = \Theta(\lg n)$.

Опростяваме г) като извадим равенствата, дефиниращи $T(n)$ и $T(n-1)$:

$$T(n) = \sum_{i=0}^{n-1} T(i) + 2^n$$

$$T(n-1) = \sum_{i=0}^{n-2} T(i) + 2^{n-1}$$

Получаваме $T(n) - T(n-1) = T(n-1) + 2^{n-1}$ или $T(n) = 2T(n-1) + 2^{n-1}$. Полученото еквивалентно отношение решаваме лесно с метода на характерисичтното уравнение. Представяме го във вида $T(n) = 2T(n-1) + \frac{1}{2}2^n$. Хомогенната част поражда уравнението $x = 2$, с множество от корени $\{2\}$. Нехомогенната част поражда множеството корени $\{2\}$. Като слеем двете мултимножества, получаваме пълния списък от корени $\{2, 2\}$. Базисните решения са $2^n, n2^n$. Последното расте най-бързо, следователно $T(n) = \Theta(n2^n)$.

Задача 2 (а) Забелязваме, че сложността на изпълняваната програма и стойността на променливата s, която накрая се връща като резултат не се различават съществено, тъй като всяко инкрементиране на s съответства на краен брой елементарни операции при изчисляване на функцията $f(n)$. Следователно без ограничения на общността считаме, че $T(n) = \Theta(f(n))$.

При всяко преминаване през цикъла for променливата s се инкрементира между $n/k - 1$ и n/k пъти. Цикълът $while$ повтаря това действие за стойности на $k = 1 \ldots n$, следователно, можем да ограничим $f(n)$ с неравенствата:

$$f(n) \leq \sum_{k=1}^{n} \frac{n}{k}$$

$$f(n) \geq \sum_{k=1}^{n} (\frac{n}{k} - 1) = \sum_{k=1}^{n} \frac{n}{k} - n$$

Оценяме асимптотично сумата в дясната страна (с H_n означаваме сумата на хармоничния ред):

$$\sum_{k=1}^{n} \frac{n}{k} = n \sum_{k=1}^{n} \frac{1}{k} = nH_n = \Theta(n \lg n)$$

От неравенствата за $f(n)$ и свойствата на асимптотичните нотации следва $T(n) = \Theta(f(n)) = \Theta(n \lg n)$.

(b) Забелязваме, че сложността на изпълняваната програма и стойността на променливата t, която накрая се връща като резултат не се различават съществено, тъй като всяко инкрементиране на t съответства на краен брой елементарни операции при изчисляване на функцията $h(n)$ или нейни рекурсивни извиквания. Следователно без ограничения на общността считаме, че $T(n) = \Theta(h(n))$.

В тривиалния случай $n < 1$ стойността е $h(n) = 1$.

Когато $n \geq 1$ стойността на $h(n)$ е сумата $h(n-1) + h(n-2) + \sum_{k=0}^{n} \binom{n}{k}$.

От развитието на израза $(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k}$ и последващо заместване на x и y с единици следва, че $2^n = (1+1)^n = \sum_{k=0}^{n} \binom{n}{k}$.

От горните разсъждения получаваме рекурентно отношение за стойността на функцията $h(n)$:

$$h(n) = h(n-1) + h(n-2) + 2^n$$

Последното решаваме лесно с метода на характериситчното уравнение. Хомогенната част поражда уравнението $x^2 = x+1$ с множество от корени $\{\frac{1-\sqrt{5}}{2}, \frac{1+\sqrt{5}}{2}\}$. Нехомогенната част поражда множеството корени $\{2\}$. Като слеем двете мултимножества, получаваме пълния списък от корени $\{\frac{1-\sqrt{5}}{2}, \frac{1+\sqrt{5}}{2}, 2\}$. Базисните решения са $(\frac{1-\sqrt{5}}{2})^n, (\frac{1+\sqrt{5}}{2})^n, 2^n$. Последното расте най-бързо, следователно $T(n) = \Theta(h(n)) = \Theta(2^n)$.

Задача 3 (a) Всички ребра на G_2 са ребра и на G_1, следователно T_2 е покриващо дърво и в G_1. T_2 ще има по-голяма сума от тегла на ребрата от T_1, защото T_1 е минимално покриващо дърво в G_1. Следователно $a_1 \leq a_2$.

(b) Нека G_1 има три върха – v_1, v_2, v_3 и три ребра, свързващи всички двойки върхове (рисунката на графа е триъгълник). Нека ребра (v_1, v_2) и (v_2, v_3) имат тегло 1, а ребро (v_1, v_3) – тегло 3. G_2 получаваме от G_1 като махнем реброто (v_1, v_3). И в двата графа минималното покриващо дърво се състои от ребрата (v_1, v_2) и (v_2, v_3) и има тегло 2.

(c) Нека G_1 е графът от подточка (b). G_2 получаваме от G_1 като махнем реброто (v_1, v_2). В G_2 минималното покриващо дърво се състои от ребрата (v_1, v_3) и (v_2, v_3) и има тегло 4, а MST на G_1 е с тегло 2.

Задача 4 Сортираме топологично G и прилагаме динамично програмиране за решаване на задачата.

Създаваме масив $C[1 \ldots n]$, като $C[i]$ ще представя броят пътища от s до i-тия връх в топологичната наредба на върховете.

Нека върхът s и t са на позиции i_0 и i_1 в топологичната наредба. Присвояваме:
$C[i] \leftarrow 0$ за $i < i_0$ (няма пътища от s до тези върхове)
$C[i] \leftarrow 1$ за $i = i_0$ (има един път от s до себе си).

За върховете с номер $i > i_0$ правим нарастващ цикъл, в който присвояваме:

$$C[i] \leftarrow \sum_{j \in Adj^{-1}(i)} C[j]$$

Смисълът на горното присвояване е, че броят на пътищата до връх i е сума на бройките пътища, достигащи предшествениците на i. С $Adj^{-1}(i)$ сме означили множеството от върховете, от които излизат ребра към i.

След изчисляването на стойностите на масива C, броят пътища до t е записан в $C[i_1]$.

Задача 5 (a) Тъй като ребрата в съчетанието M не се допират, всяко от тях трябва да бъде покрито от различен връх от върховото покритие C. За всяко ребро $(u, v) \in M$ ще бъде вярно $(u \in C) \vee (v \in C)$, следователно $|M| \leq |C|$.

(b) Нека M е максимално съчетание. Нека C_M е множеството от върхове, които са краища на ребрата от M. Очевидно $|C_M| = 2|M|$.

C_M е върхово покритие. Ако допуснем, че съществува ребро (u, v), непокрито от C_M, следва че това ребро не се допира до никое ребро от M и можем да получим по-голямо съчетание $M \cup \{(u, v)\}$, което противоречи на максималността на M.

Щом C_M е върхово покритие, а C е минимално върхово покритие, вярно е неравенството $|C_M| \geq |C|$. заместваме $|C_M|$ с $2|M|$ и получаваме $2|M| \geq |C|$.

Задача 6 Редукция 1 не е коректна. Да разгледаме индивидуална задача на SAT, която се състои от 2 еднобуквени дизюнкции (y) и $(\overline{y})$. Очевидно тя не може да бъде удовлетворена. Съгласно правилата на редукцията двете дизюнкции ще бъдат преобразувани в две трибуквени дизюнкции $(y \vee w_1 \vee w_2)$ и $(\overline{y} \vee w_3 \vee w_4)$, които могат да бъдат удолетворени от присвояване на значения $w_1 = TRUE$ и $w_3 = TRUE$.

Редукция 2 е коректна. 3SAT е частен случай на SAT. Входната индивидуална задача на 3SAT че има удовлетворяващ набор от присвоявания на променливите точно когато същият набор присвоявания удовлетворява същата задача, но разгледана като индивидуална задача на SAT.