نمونه سوال طراحی الگوریتم

سوالات فرد ۱-۹۸-۹۷

فاطمه طوسی مفتاح ۹۷۰۱۴۲۴۳۴۲

۱-۹۸-۹۷

تستی

۱- بهترین حالت زمان اجرای الگوریتم مرتب سازی درجی (Insertion sort) زمانی رخ میدهد که ...

۱. داده‌های ورودی مسئله، خود از قبل مرتب شده باشند ✓

۲. داده‌های ورودی مسئله، برعکس مرتب شده باشند.

۳. داده‌های ورودی مسئله، به صورت یکی درمیان مرتب باشند.

۴. در الگوریتم مرتب سازی درجی، هیچ حالت بهترینی وجود ندارد.

بهترین حالت زمانی است که آرایه از قبل مرتب شده باشد در این حالت زمان اجرای الگوریتم از $O(n)$ است.

۳- فرض کنید $T_1(n)$ و $T_2(n)$ زمان اجرای دو قطعه برنامه P_1 و P_2 باشند و داریم: $T_1(n) \in O(f(n))$ و $T_2(n) \in O(g(n))$

مقدار $T_1(n) + T_2(n)$ زمانی که قطعه برنامه P_2 در راستای قطعه برنامه P_1 اجرا می‌شود برابر است با: ۱. $O(\min\{f(n), g(n)\})$ ۲. $O(\max\{f(n), g(n)\})$

۳. $O(f(n) + (g(n)))$ ۴. $O(f(n) \cdot g(n))$

می‌دانیم که $T_1(n) \in O(f(n))$ بنابراین c_1 و n_1 وجود دارد که برای:

$\forall n \geq n_1 \quad T_1(n) \leq c_1 f(n)$

و همچنین $T_2(n) \in O(g(n))$ بنابراین c_2 و n_2 وجود دارد که برای:

$\forall n \geq n_2 \quad T_2(n) \leq c_2 g(n)$

$\Rightarrow T_1(n) + T_2(n) \leq c_1 f(n) + c_2 g(n)$

$\in (c_1 + c_2) \max\{f(n), g(n)\}$

که در آن با انتخاب $n_0 = \max\{n_1, n_2\}$ خواهیم داشت:

$T_1(n) + T_2(n) \in O(\max\{f(n), g(n)\})$

نعم

۵- کدام گزینه رابطه بازگشتی محاسبه زمان اجرای الگوریتم ضرب ماتریس ها به روش استراسن را نشان می دهد؟

۱. $\begin{cases} T(1)=1 \\ T(n)=7T(\frac{n}{2})+18T(\frac{n}{2})^2 \end{cases}$

۲. $\begin{cases} T(1)=1 \\ T(n)=8T(\frac{n}{2})+14(\frac{n}{2})^2 \end{cases}$

۳. $\begin{cases} T(1)=1 \\ T(n)=7T(\frac{n}{2})+18(\frac{n}{2})^2 \end{cases}$ ✓

۴. $\begin{cases} T(1)=1 \\ T(n)=8T(\frac{n}{2})+17(\frac{n}{2})^2 \end{cases}$

$$T(n)=7n^{\log 7}-6n^2 \Rightarrow T(n)\in\Theta(n^{2.81})$$

۷- در جستجوی دودویی لیست زیر، در صورتی که به دنبال یافتن عدد ۷۱ در لیست باشیم پس از چند مقایسه به نتیجه NotFound (پیدا نشد) خواهیم رسید؟

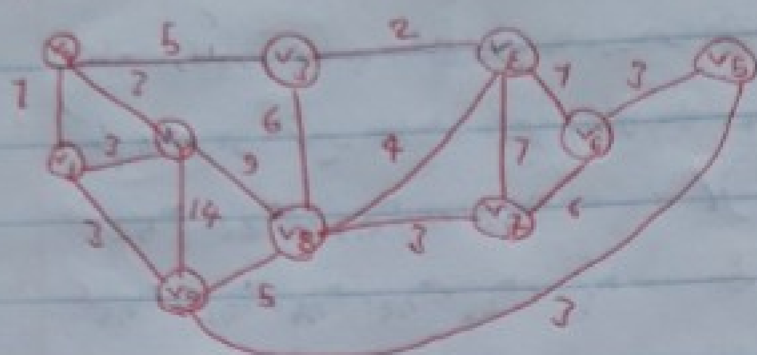
اندیس	۰	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲
مقدار	۳	۹	۱۲	۲۷	۴۳	۴۹	۴۸	۴۹	۵۴	۶۰	۸۱	۹۸	۱۲۰

۱. 2 مقایسه   ۲. 3 مقایسه   ۳. 4 مقایسه ✓   ۴. 5 مقایسه

لیست به دو زیر لیست تقسیم می کنیم (با طول برابر) اگر عدد مسئله (x) کوچکتر از عنصر میانی باشد زیر لیست چپ انتخاب می شود، در غیر این صورت زیر لیست راست. اگر عنصر x را در لیست جدید در صورتی که لیست به اندازه کافی کوچک باشد جستجو می کند. در غیر اینصورت عمل تقسیم لیست به دو لیست کوچکتر دوباره انجام می شود و در این مسئله تا 4 مقایسه لازم می باشد تا به این نتیجه برسیم که ۷۱ پیدا نشد.

۹- با در نظر گرفتن گراف مقابل و با استفاده از الگوریتم کروسکال، هشتمین یالی که به درخت پوشای مینیمم حاصل اضافه می شود، کدام یال است؟

۱. یال $v_9 - v_2$ ۲. یال $v_9 - v_8$ ۳. یال $v_8 - v_4$ ✓   ۴. یال $v_2 - v_5$



$F = \{v_{51}, v_{53}, v_{54}, v_{12}, v_{19}, v_{29}, v_{28}, v_{38}, v_{48}, v_{44}, v_{48} \ldots\}$

۱۱- با در نظر گرفتن اشیاء زیر و هم چنین کوله پشتی با ظرفیت ۴۰ کیلوگرم، حداکثر ارزش حاصل از مسئله کوله پشتی (غیر صفر و یک - کسری) با استفاده از اشیاء موجود در جدول برابر خواهد بود با:

شماره کالا	۱	۲	۳	۴	۵
ارزش	۸	۵	۱۵	۱۰	۲۰
وزن	۱۶	۱۵	۲۵	۸	۱۵

۱. 44   ۲. 38.5

۳. 40.2 ✓   ۴. 40.9

مرحله اول: شیء با بالاترین اولویت عنصر شماره ۵ می‌باشد انتخاب می‌شود و M برابر با ۲۵ است. مرحله دوم: عنصر با بالاترین اولویت که عنصر ۴ است انتخاب می‌شود و مقدار M برابر با ۱۷ است. مرحله سوم: عنصر x_3 بالاترین اولویت را دارا است، ولی وزن آن که ۲۵ می‌باشد از ظرفیت باقیمانده کوله پشتی که برابر ۱۷ است بیشتر است. بنابراین حلقه for شکسته می‌شود و شرط if در الگوریتم اجرا می‌شود. در نهایت خروجی زیر بدست می‌آید:

x_i	x_1	x_2	x_3	x_4	x_5
	۰	۰	$\frac{17}{25}$	۱	۱

ارزش یا سود حاصل از کوله پشتی به صورت زیر انتخاب می‌کنیم:

$$\sum_{i=1}^{5} p_i w_i = 8\times 0 + 5\times 0 + 15\times \frac{17}{25} + 10\times 1 + 20\times 1 = 40.2$$

جواب بهینه $(1, 1, \frac{17}{25}, 0, 0)$ x_s می‌باشد.

۱۴- در الگوریتم محاسبه حداقل ضرب ها در زنجیره ضرب ماتریسها، برای محاسبه $m_{1,4}$ نیاز به داشتن کدام مقادیر در ماتریس محاسبات داریم. (برای محاسبه $m_{1,4}$ از کدام مقادیر ماتریس استفاده خواهیم کرد)

۱- $m_{1,2}, m_{2,2}, m_{2,3}, m_{1,3}$

۲- $m_{4,4}, m_{1,3}, m_{3,4}, m_{1,2}, m_{2,4}, m_{1,1}$ ✓

۳- $m_{4,4}, m_{3,3}, m_{3,4}, m_{2,2}, m_{2,4}, m_{1,1}$

۴- $m_{4,4}, m_{3,3}, m_{3,4}, m_{2,2}, m_{2,3}, m_{1,2}$

۱۵- پیچیدگی زمانی الگوریتم بازگشتی تور بهینه در یک گراف (مسئله فروشنده دوره گرد) برابر با کدام گزینه است؟

1- $\theta(n2^{n})$ 2- $\theta(n^2 2^n)$ ✓ 3- $\theta(2^n)$ 4- $\theta(n^2 \log n)$

و وقتی عملی است که n کوچک باشد.

۱۷- کدام گزینه سود بهینه حاصل از انتخاب i شیء (قطعه) اول به شرطی که وزن کل از w بیشتر نشود را، روش برنامه نویسی پویا (برای حل مسئله کوله پشتی) نشان می دهد.

۱- $P[i][w] = \begin{cases} \text{maximum}(P[i][w-1], P[i-1][w-w_i]) & \text{if } w_i \le w \\ P[i-1][w] & \text{if } w_i > w \end{cases}$

۲- $P[i][w] = \begin{cases} \text{maximum}(P[i-1][w], P_i + P[i-1][w-w_i]) & \text{if } w_i \le w \\ P[i-1][w] & \text{if } w_i > w \end{cases}$ ✓

۳- $P[i][w] = \begin{cases} \text{maximum}(P[i][w-1], P_i + P[i-1][w-w_i]) & \text{if } w_i > w \\ & \text{if } w_i \le w \end{cases}$

۴- $P[i][w] = \begin{cases} \text{maximum}(P[i-1], P_i + P[i-1][w-w_i]) & \text{if } w_i > w \\ P[i-1][w] & \text{if } w_i \le w \end{cases}$

max اگر w در w انتخاب، $P[i][w] = P[i-1][w]$ خواهد بود. یعنی اگر وزن قطعه i ام بیشتر از وزن کل قابل تحمل کوله پشتی باشد، آن قطعه را برای قرار دادن در کوله پشتی انتخاب نمی کنیم و سود بهینه حاصل از انتخاب i قطعه اول برابر سود بهینه حاصل از انتخاب i-1 قطعه اول خواهد بود.

Persian Note Book

Subject:

Date:

۱۹- کدام یک از موارد زیر صحیح است؟ مورد اول: مسئله‌ای که به روش بازگشت به عقب حل شود، می‌تواند بیش از یک جواب داشته باشد و هیچ جوابی بر جواب دیگر امتیازی ندارد.
مورد دوم: در اغلب مسائلی که به روش انشعاب و تحدید حل می‌شوند، هم یافتن جواب بهینه است.
مورد سوم: الگوی جستجو در درخت برای روش انشعاب و تحدید، جستجوی عمقی است.

۱. فقط موارد اول و دوم ۲. موارد دوم و سوم ۳. موارد اول و سوم ۴. موارد اول و دوم و سوم

گزینه‌ی ۱

۲۱- برای مجموعه کارهای زیر، با سود و مهلت داده شده، بیشترین سودی که می‌توان کسب نمود برابر است با ...
(مسئله زمان‌بندی با مهلت)

کار	۱	۲	۳	۴	۵	۶	۷	۸
سود	۸۹	۷۴	۶۹	۴۲	۵۹	۱۶	۱۹	۱۲
مهلت	۳	۱	۴	۲	۳	۲	۳	۳

۱. ۱۲۸ ۲. ۱۳۵

۳. ۲۹۱ ✓ ۴. ۲۷۴

۲۳- کدام گزینه رابطه بازگشتی مربوط به الگوریتم‌ها ضرب دو عدد بزرگ n رقمی را به درستی بیان می‌کند؟

۱. $T(n)=2T(\frac{n}{4})+cn$ ۲. $T(n)=4T(\frac{n}{2})+cn^2$

۳. $T(n)=2T(\frac{n}{4})+cn^2$ ۴. $T(n)=4T(\frac{n}{2})+cn$ ✓

با فرض $n=2^k$ خواهیم داشت: $T(2^k)=4T(2^{k-1})+c2^k$ و یا $T(k)=4T(k-1)+c2^k$

با این معادله می‌توان گفت: $T(k)=c_1 4^k+c_2 2^k$ با جایگذاری n به جای 2^k رابطه زیر حاصل می‌شود:

$T(n)=c_1n^2+c_2n$ بنابراین: $T(n)\in\Theta(n^2)$ و پیچیدگی الگوریتم‌ها از درجه ۲ است چون این الگوریتم از چهار عمل ضرب در محاسبات خود استفاده می‌کند. اگر تعداد این ضرب‌ها را کاهش داد الگوریتم بهتری می‌توان ساخت.

Persian Note Book

Subject:

Date:

۲۵- کدام یک از موارد در خصوص مسائل تصمیم‌گیری درست است؟

مورد اول: مسائل NP زیرمجموعه مسائل P هستند. مورد دوم: مسائل P زیرمجموعه مسائل NP هستند. مورد سوم: مسائل تصمیم‌گیری‌ای وجود دارند که نه NP هستند و نه P. مورد چهارم: همه مسائل تصمیم‌گیری یا از نوع P هستند یا از نوع NP.

۱. موارد اول و دوم   ۲. موارد دوم و سوم ✓

۳. موارد سوم و چهارم   ۴. موارد اول و چهارم

در بیان یک الگوریتم نامعین ترتیبی ندارد از هم دستورها و تابع‌های دیگر نباید استفاده شود. بنابراین هر الگوریتم معین توسط یک کامپیوتر نامعین قابل اجرا است پس $P \subseteq NP$. برای مورد سوم باید ثابت کنیم که طراحی یک الگوریتم زمانی چندجمله‌ای برای آن غیر ممکن است و برای اینکه در NP باشد باید الگوریتمی وجود داشته باشد که عمل تصدیق را در زمان چند[جمله‌ای انجام دهد].

تشریحی

۱- رابطه بازگشتی زیر را حل نمایید. $T(n)=3T(n-1)+4T(n-2)$ جایگزینی انجام دهید

$T(0)=0,\ T(1)=1$

$$T(n)=x^n \xrightarrow{\text{جایگزینی}} x^n = 3x^{n-1} + 4T(n-2) \Rightarrow n^2 - 3n + 4 = 0 \Rightarrow n_1 = -1 \quad n_2 = 4$$

بنابراین $\rightarrow T(n) = c_1(-1)^n + c_2 4^n$

$$\begin{cases} c_1 + c_2 = 0 \\ -c_1 + 4c_2 = 1 \end{cases} \Rightarrow c_1 = -\frac{1}{5}, \quad c_2 = \frac{1}{5}$$

$$T(n) = \frac{1}{5}(4^n - (-1)^n) \Rightarrow T(n) \in O(4^n)$$

۳- فرض کنید متنی شامل حروف a, b, c, d, e, f, g و h باشد. تعداد کاراکترهای این متن برابر 519 کاراکتر است که فراوانی تعداد تکرار کاراکترها به صورت زیر می‌باشد.

الگوریتم کدگذاری هافمن را بر روی این کاراکترها اعمال نموده و درخت کدگذاری را مرحله به مرحله رسم نموده و در نهایت کدهای مربوط به حروف را استخراج نمایید.

حرف	a	b	c	d	e	f	g	h
تکرار	۶	۴۱	۲۹	۴۳	۱۲۴	۲۶	۶۴	۱۵۸
کد								

مرحله اول: مشکل اولیه که در آن مشکل بر اساس تعداد کدها به صورت صعودی مرتب شده اند را ارائه می دهیم

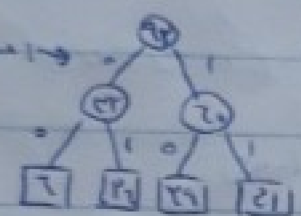
h:۱۵۸ e:۱۲۴ g:۱۰۳ d:۴۲ b:۲۱ c:۲۹ f:۳۶ a:۶

مرحله دوم: ادغام دو درخت با کمترین مقدار و سپس مطابق الگوریتم بالا و ارائه لیست مرتب حاصل به صورت غیر نزولی:

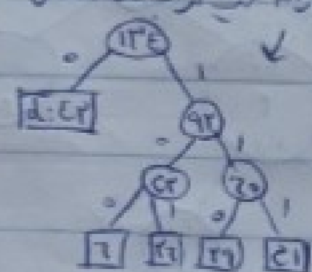
مرحله سوم: ادغام دو درخت با کمترین مقدار و ارائه لیست مرتب شده حاصل به صورت غیر نزولی:

مرحله چهارم: ادغام دو درخت با کمترین مقدار و ارائه لیست مرتب شده حاصل به صورت غیر نزولی:

به مرحله چهارم

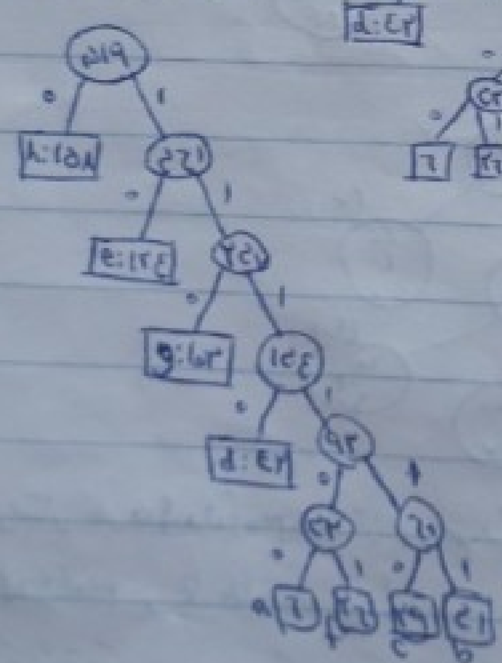


مرحله پنجم: ادغام دو درخت با کمترین مقدار و ارائه لیست مرتب شده حاصل به صورت غیر نزولی:



و به این ترتیب موارد ادغام را انجام می دهیم

: مرحله آخر



کاراکترها	کدها
a	۱۱۰۰
f	۱۱۰۱
c	۱۱۱۰
b	۱۱۱۱۱۱
d	۱۱۱۰
g	۱۱۰
e	۱۰
h	۰

۵- فرض کنید کالاهای زیر را داریم:

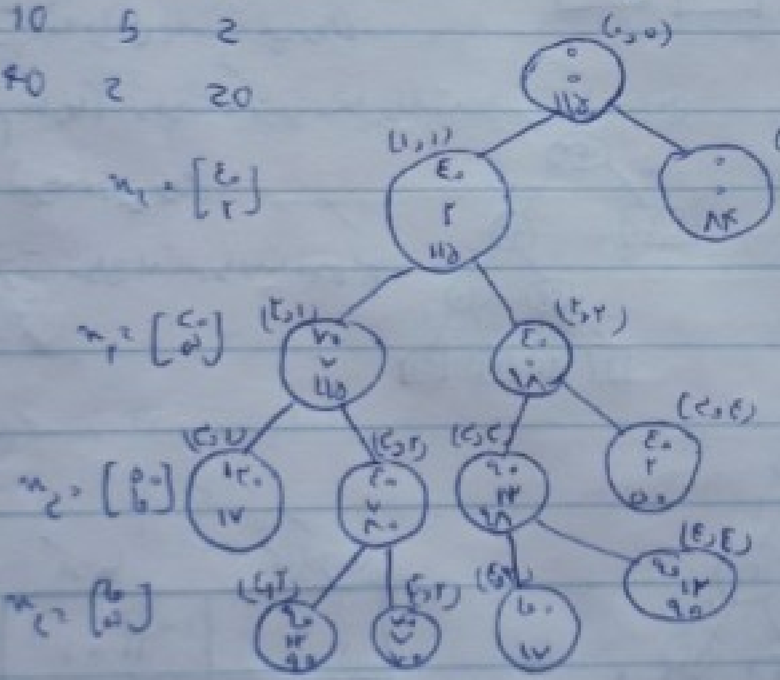
شماره کالا	۱	۲	۳	۴
ارزش	۵۰	۳۰	۱۰	۴۰
وزن	۱۰	۵	۵	۲

اگر ظرفیت کوله پشتی برابر ۱۶ کیلوگرم باشد. مسئله کوله پشتی صفر و یک با الگوریتم تکنیک شاخه و کران حل نمایید. درخت فضای جستجو را به طور کامل رسم نمایید و در نهایت حداکثر سود ممکن را محاسبه نمایید.

i	p_i ارزش	w_i وزن	p_i/w_i ارزش/وزن
1	50	10	5
2	30	5	6
3	10	5	2
4	40	2	20

فرض کنید $n=4$ و $W=16$ داشته باشیم:

در ابتدا قطعه‌ها را براساس ارزش/وزن به صورت غیر نزولی مرتب می‌کنیم:



در ابتدا مقدار $max\ profit = \$0$ قرار می‌دهیم. اگر ریشه را مشخص می‌کنیم ... ارزش و وزن آن profit = $0 و weight = 0 می‌شود. برای محاسبه حد ابتدا k را محاسبه می‌کنیم. k مکان ... سطحی است که اگر آن گره را به جمع وزن گره‌های قبلی اضافه کنیم حاصل جمع بیشتر از ۱۶ خواهد شد. چون ۱۷=۱۰+۵+۲ و ۱۷ از ۱۶ بزرگتر است پس k=۳ سوم حاصل جمع اوزان را ... داریم:

$$total\ weight = weight + \sum_{j=i+1}^{k-1} w_j = 0 + 2 + 5 = 7$$

$$bound = profit + \sum_{j=i+1}^{k-1} p_j + (W - total\ weight) \times \frac{p_k}{w_k}$$

$$= \$0 + \$40 + \$30 + (16 - 7) \times \$\frac{50}{10} = \$115$$

Persian Note Book

Subject:

Date:

چون همه با گره ۵ همراه است و کمتر از وزن تحمل کوله پشتی یعنی ۱۶ است. پس کران بالای چنین است که ۱۱۵ $ بزرگتر از $۰ یعنی مقدار maxprofit است.

۶. گره (۱،۱) را بررسی می‌کنیم. ارزش و وزن گره $profit = \$0 + \$40 = \$40$, $weight = 0 + 2 = 2$ چون که وزن آن ۲ است و کوچکتر از ۱۶ است و ارزش آن $۴۰ است بزرگتر از $۰ است و maxprofit برابر $۴۰ قرار داده می‌شود. حد آن را محاسبه می‌کنیم چون ۱۷ = ۱۰ + ۵ + ۲ و ۱۷ از W = ۱۶ بزرگتر است قطعه سوم حاصل جمع اوزان را از W بزرگتر می‌رود. بنابراین k = ۳

$$tw = w + \sum_{j=i+1}^{k-1} w_j = 2 + 5 = 7$$

$$\$40 + \$30 + (16 - 7) \times \frac{\$50}{10} = \$115$$

چون وزن آن ۲ است و کوچکتر از W = ۱۶ و ارزش آن ۴۰ بزرگتر از $۰ maxprofit برابر $۴۰ خواهد بود

و به همین ترتیب تمامی گره‌ها را حل می‌نماییم

در درخت فضای حالت هرس شده فقط ۱۳ گره وجود دارد در حالی که کل درخت فضای حالت ۳۱ گره دارد

نمونه سوال طراحی الگوریتم

تستی

سوالات نوبت ۱-۹۵-۹۴

Date: ۹۷/۱۲/۲۳ نام استاد: علوی‌منش ۹۴-۹۵-۱ Subject:

۲- کدام ترتیب مقایسه‌ای صحیح بین پیچیدگی زمانی الگوریتم‌ها را نشان می‌دهد

۱. ✓ $O(\sqrt{n}) < O(n) < O(n\log n)$ ۲. $O(3^n) < O(n!) < O(n^n)$

۳. $O(n) < O(n\log n) < O(\sqrt{n})$ ۴. $O(n\log n) < O(n^3) < O(n^2\log n)$

۴- در مورد توابع زیر کدام ترتیب صحیح می‌باشد؟

۱. $O(n\log n), O((1+\epsilon)^n), O(\frac{n^2}{\log n})$ ۲. $O((1+\epsilon)^n), O(n\log n), O(\frac{n^2}{\log n})$

۳. $O(\frac{n^2}{\log n}), O(n\log n), O((1+\epsilon)^n)$ ۴. ✓ $O(n\log n), O(\frac{n^2}{\log n}), O((1+\epsilon)^n)$

۶- جواب رابطه بازگشتی زیر کدام است؟ $T(n)=T(\frac{n}{3})+T(\frac{2n}{3})+O(n)$

۱. $O(n)$ ۲. ✓ $O(n\log n)$ ۳. $O(n^2\log n)$ ۴. $O(n^2\sqrt{n})$

مشاهده می‌کنید که از بین شاخه‌ها طولانی‌ترین اجرا شده همان را حساب می‌کنیم

$T(n)$

مجموع n ← $T(\frac{n}{3})$ $T(\frac{2n}{3})$

$T(\frac{n}{9})$ $T(\frac{2n}{9})$ $T(\frac{2n}{9})$ $T(\frac{4n}{9})$

$T(\frac{2^k n}{3^k})$

چون باید به $T(1)$ برسد پس:

$(\frac{2}{3})^k \times n = 1 \Rightarrow (\frac{2}{3})^k = \frac{1}{n} \Rightarrow n = (\frac{3}{2})^k \Rightarrow k = \log_{\frac{3}{2}} n$

چون k تا سطح موجود است پس:

$T(n) \in nk = n\log_{\frac{3}{2}} n \Rightarrow T(n) \in O(n\log n)$

۸- بدترین حالت زمانی الگوریتم جستجوی دودویی برای جستجوی موفق و ناموفق به ترتیب از راست به چپ کدام است؟

۱. $O(\log n), O(\log n)$ ۲. ✓ $\theta(\log n), O(\log n)$

۳. $O(\log n), \theta(\log n)$ ۴. $\theta(\log n), \theta(\log n)$

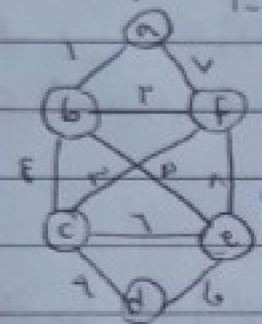
درخت تصمیم دودویی را برای عملگر BinSearch روی n عنصر در نظر بگیرید. در این درخت تمام جستجوهای موفق در یک گره داخلی صورت می‌گیرد و تمام جستجوهای ناموفق در یک گره خارجی خاتمه می‌یابد. اگر $2^{k-1} \le n < 2^k$ باشد آنگاه همه گره‌های داخلی در سطح ۱، ۲، ...، k قرار می‌گیرند و همه گره‌های خارجی در سطح k و k+1 قرار می‌گیرند. تعداد مقایسه‌های لازم برای خاتمه الگوریتم در یک گره خارجی در سطح i برابر i است و در گره داخلی در سطح i برابر i-1. بنابراین $k = \log n$ یا $k-1$

$T(n) \in \begin{cases} O(\log n) & \text{جستجوی موفق} \\ \theta(\log n) & \text{ناموفق} \end{cases}$

Date: Subject:

۱۲- در ضرب ماتریس‌ها به روش استراسن اگر مسائل کوچک، ضرب ماتریس ۲×۲ باشد، برای ضرب دو ماتریس ۸×۸ چند ضرب عددی صورت می‌پذیرد؟ ۱. ۳۹۲ ✓ ۲. ۳۴۳
۳. ۵۱۲ ۴. ۲۵۶

۱۳- در گراف زیر با اجرای الگوریتم پرایم و شروع از رأس a، درخت پوشای مینیمم دارای کدام هزینه خواهد بود؟
۱. ۱۱ ۲. ۱۵ ۳. ۲۰ ✓ ۴. ۲۲



طبق الگوریتم ابتدا راس $Y=\{v_a\}$ انتخاب می‌شود و $a \in F$ خواهد بود. حال رئوس مجاور Y ها را بررسی و به ترتیب از کوچکترین انتخاب می‌کنیم و در مرحله آخر چون $Y=V$ می‌باشد. بنابراین گراف حاصل به عنوان درخت پوشای مینیمم با مقدار هزینه ۲۰ می‌باشد.

۱۴- در هر مسئله گراف خلوت (متراکم) با شد الگوریتم ... سریع‌تر از الگوریتم ... عمل می‌کند و در این حالت پیچیدگی زمانی الگوریتم کروسکال ... است. (به ترتیب از راست به چپ)
۱. کروسکال، پرایم، $\Theta(n\log n)$ ✓ ۲. کروسکال، پرایم، $\Theta(n)$
۳. پرایم، کروسکال، $\Theta(n)$ ۴. پرایم، کروسکال، $\Theta(n^2\log n)$

با توجه به رابطه $(n-1) \le m \le n(n-1)/2$ و رابطه الگوریتم پرایم $T(n) \in \Theta(n^2)$
و الگوریتم کروسکال $T(n) \in \Theta(n^2\log n)$ و $T(m) \in \Theta(m\log m)$

~~۱۵~~ ۱۶- فرض کنید برای $n=7$ کارها، مهلت و بهره‌ها و مربوط به کارها را به صورت زیر داریم. جواب بهینه با الگوریتم زمان‌بندی با مهلت کدام است؟
۱. جواب بهینه {1,2,6,9} با سود 130 ۲. جواب بهینه {2,4,1,5} با سود 130
۳. جواب بهینه {2,4,1} با سود 170 ۴. جواب بهینه {2,4,7,1} با سود 130 ✓

پاسخ صفحه بعد

نخست L را برابر صفر قرار می‌دهیم.
مقدار

سود	مهلت	کار
۶۵	۴	۱
۵۰	۱	۲
۳۵	۱	۳
۳۰	۲	۴
۱۵	۴	۵
۱۰	۱	۶
۵	۲	۷

مجموعه امکان‌پذیر	سود	J	مرحله
			۰
هست	۰	∅	۱
هست	۶۵	{۱}	۲
هست	۱۱۵	{۲,۱}	۳
نیست	۱۱۵	{۲,۳,۱}	۴
هست	۱۴۵	{۲,۴,۱}	۵
نیست	۱۴۵	{۲,۴,۱,۵}	۶
نیست	۱۴۵	{۱,۲,۴,۶}	۷
نیست	۱۴۵	{۲,۴,۷,۱}	

پاسخ می ۱۴۵

حل بهینه
{۱,۲,۴} حل
باید ۳ کار باشد

۲۴- [illegible] درخت‌های جستجوی دودویی [illegible] کدام است؟

۱. $O(n!)$ ۲. $O(2^n)$ ۳. $O(n^2 \log n)$ ۴. $O(n^3)$ ✓

برای انجام این کار سه حلقه تودرتو داریم (با اندیس‌های i و j و k) چون (i و j و k) است پس از این مقادیر معلوم

ما و تعداد تکرارها از حلقه k عبارت است از: $j-1 = i+1-1 = i$ ، $i+1-1-i+1 = 1$

Date: چون ما اندیس ... برابر با n-1 است. ... از حلقه for با اندیس ... تعداد تکرارها Subject:

با این مقادیر ... [illegible]

[illegible]	[illegible]	[illegible]
[illegible]	3	50
[illegible]	1	50
[illegible]	1	40
[illegible]	[illegible]	30
5	3	75
[illegible]	1	70

T اسم تغییر می‌کند، تعداد کل عملیات‌هایی که عمل اصلی انجام می‌شود عبارت است از:

$$\sum_{i=1}^{n-1} [(n-1) \times i]$$

که این عبارت برابر است با:

$$\frac{n(n-1)(n+1)}{2} \in \Theta(n^2)$$

۱۸- تعداد اعمال جمع برای الگوریتم ضریب دوجمله‌ای $\binom{5}{3}$ با استفاده از برنامه‌نویسی پویا کدام است؟

۱. ۶ ۲. ۹ ✓ ۳. ۱۲ ۴. ۱۹

$T(n,k) = k(2n-k-1)/2$

$$\binom{5}{3} = \frac{3(2\times 5-3-1)}{2} = 9$$

۲۰- پیچیدگی محاسباتی در حالت برای الگوریتم حاصل‌ضرب‌ها ... می‌باشد:

۱. $\Theta(n^2 2^n)$ ۲. $\Theta(n\log n)$ ۳. $\Theta(n^2)$ ۴. $\Theta(n^3)$ ✓

بالای صفحه

۲۲- تعداد فراخوانی‌ها برای محاسبه $p(3,3)$ در تابع بازگشتی world series زیر کدام است؟

۱. ۲۰ ۲. ۱۸ ۳. ۴۰ ۴. ۲۸ ✓

```
float worldseries(int n, float p, float q)
{ int m,k;
  float p[][n+1];
  for (m=1; m<=n; m++)
  {
   p[0][m]=1;
   p[m][0]=0;
   for(k=1;k<=m-1;k++)
    p[k][m-k]=p*p[k-1][m-k]+q*p[k][m-k-1];
  }
  for(m=1; m<=n; m++)
   for(k=0; k<=n-m;k++)
    p[m+k][n-k]=p*p[m][m+k-1]+q*p[k+m][n-k-1];
   return p[n][n];
}
```

Nick Andish

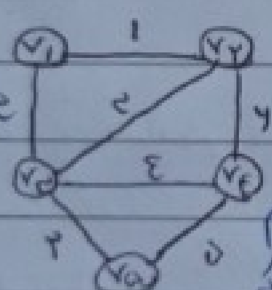
{۱و۷و۴و۲} ۱۴۰

۲۴- تعداد درخت‌های جستجوی دودویی که با n عنصر متمایز می‌توان ساخت کدام است؟

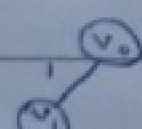
۱. $O(n!)$ ۲. $O(2^n)$ ۳. $O(n^2 \log n)$ ۴. ✓ $O(n^n)$

تشریحی

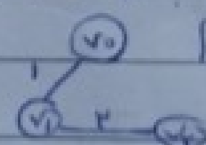
۳- الگوریتم کروسکال را برای گراف زیر اجرا کنید، درخت پوشای مینیمم را مرحله به مرحله رسم کرده و هزینه‌ی نهایی درخت حاصل را بدست آورید؟



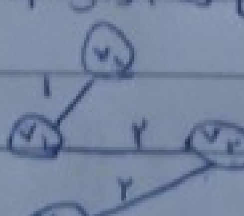
نخست درخت heap برای مرتب کردن یال‌ها برحسب وزن یال‌ها تشکیل می‌شود که در ریشه‌ی درخت heap، $e_{01} = 1$ قرار می‌گیرد و $F = \phi$ و ۲ مجموعه‌ی مجزا از هم تشکیل می‌شود. مرحله اول: $e_{01} = 1$ به عنوان یال با هزینه‌ی کمتر انتخاب می‌شود، زیرا v_0, v_1 و v_1, v_2 در یک مجموعه نباشند بنابراین داریم: $F\{e_{01}\}$



مرحله دوم: $e_{12} = 2$ به عنوان یال کم هزینه انتخاب می‌شود که در آن v_1, v_2 و v_2, v_4 در یک مجموعه نباشند بنابراین: $F\{e_{01}, e_{12}\}$

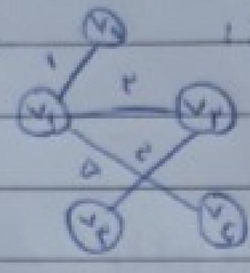


مرحله سوم: $e_{24} = 2$ به عنوان یال کم هزینه انتخاب می‌شود که در آن v_2, v_4 و v_1, v_2 در یک مجموعه نباشند بنابراین: $F\{e_{01}, e_{12}, e_{24}\}$



مرحله چهارم: یال ۴ e_{24} انتخاب می‌شود ولی چون دو راس آن در یک مجموعه قرار دارند
بنابراین با اضافه کردن آن یک دور ایجاد می‌شود. لذا از این یال برای اضافه کردن به درخت صرف نظر
می‌شود (این یال از درخت حذف می‌شود)

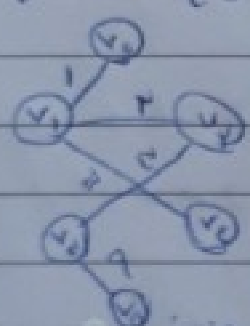
مرحله پنجم: یال ۵ e_{25} به عنوان یال با وزن کم انتخاب می‌شود که v_1, v_2 و v_2, v_7 می‌باشد
(یال e_{25} از درخت حذف می‌شود) بنابراین: $F=\{e_{51}, e_{13}$



مرحله ششم: یال ۶ e_{34} انتخاب
می‌شود ولی چون دو راس این یال در یک
مجموعه قرار دارند بنابراین با اضافه کردن آن یک
دور در درخت ایجاد خواهد شد. لذا این یال حذف می‌شود.

مرحله هفتم: یال ۷ e_{24} انتخاب می‌شود یعنی یال e_{24} در نظر گرفته می‌شود هر دو
آنها با اضافه شدن به درخت، ایجاد دور در درخت می‌کنند بنابراین حذف می‌شوند.

مرحله آخر: ۹ e_{25} یال با وزن ۹ انتخاب می‌شود که دو راس v_2, v_4 و v_5, v_7
می‌باشد بنابراین: $F=\{e_{51}, e_{13}, e_{34}, e_{16}, e_{65}$ و $e_{45}\}$



و چون تعداد یال‌های اضافه شده F برابر n-1 می‌باشد ($n-1$ یال)
بنابراین حلقه پایان می‌پذیرد و خروجی الگوریتم مجموعه F، ۶ یال
های درخت پوشای مینیمم را در خود جای داده است، می‌باشد و هزینه
کل درخت، همانطور که قبلاً محاسبه شده بود برابر ۲۱ خواهد بود.

Date: .... Subject: ....

۴- برنامه مربوط به طولانی‌ترین زیررشته مشترک دو رشته X و Y را با برنامه‌نویسی پویا بنویسید.

```
print_LCS(b, x, length(x), length(Y))
void print_LCS(b, x, i, j)
{
  if(i=0 || j=0)
    return;
  if ( b[i,j] = '↖' )
  {
    print_LCS(b, x, i-1, j-1);
    print x_i;
  }
  else if (b[i,j] = '↑')
    print_LCS(b, x, i-1, j);
  else
    print_LCS(b, x, i, j-1);
}
```

این تابع "BCBA" را چاپ می‌کند. زمان تابع O(m+n) است چون مطابق یک واحد از i یا j در هر مرحله بازگشت کم می‌شود.