



شماره دانشجویی:

بسمه تعالی
آزمون پایان ترم
ساختمان داده‌ها و الگوریتم‌ها
شنبه ۱۷ دی ۱۴۰۱

مدت آزمون ۱۲۰ دقیقه



نام و نام خانوادگی:

۱ - درستی یا نادرستی هر یک از گزاره‌های زیر را مشخص کنید (نیازی به توضیح در مورد چرایی آن نیست). (۵ نمره)

درست	نادرست	هر درخت با n راس حتما کمتر از n یال دارد.
درست	نادرست	در یک مسئله برنامه‌نویسی پویا، اگر m زیر مسئله مختلف باید حل شوند، حافظه مورد نیاز $\Omega(m)$ خواهد بود.
درست	نادرست	برای حل یک مسئله با برنامه‌نویسی پویا، هم می‌توان از روش پایین به بالا (bottom-up) استفاده کرد و هم از روش بالا به پایین (top-down). زمان اجرای هر دو روش همواره یکسان خواهد بود.
درست	نادرست	اگر در یک گراف G ، تعداد یال‌ها از تعداد راس‌ها بیشتر باشد، حتما یک دور در آن وجود دارد.
درست	نادرست	یک گراف دو بخشی می‌تواند دور زوج داشته باشد.

۲ - رابطه بازگشتی زمان اجرای یک الگوریتم به صورت زیر است. این تابع را حل کنید. (۲ نمره)

$$T(n) = 8T(n/2) + O(n^3)$$

با استفاده از قضیه اصلی خواهیم داشت: $a=8$ ، $b=2$ و $d=3$. چون $d = \log_b a = \log_2 8 = 3$ ، پس جواب به صورت $O(n^3 \log n)$ خواهد بود.

۳ - علی یک پیاده‌سازی از جدول درهم‌سازی انجام داده است که عدد ورودی x را به خانه $(5x+7) \bmod 31 \bmod n$ نگاشت می‌کند (n تعداد اعدادی است که در جدول درج خواهیم کرد). می‌خواهیم ۷ عدد زیر را به ترتیب از چپ به راست (یعنی ابتدا ۱۲، سپس ۱۳، ...) در این جدول درج کنیم:

12, 13, 6, 8, 15, 18, 3

پس از درج اعداد، مشخص کنید که در هر یک از خانه‌های جدول درهم‌ساز، کدام اعداد و به چه ترتیبی قرار خواهند گرفت. (۴ نمره)

0:	
1:	3
2:	8
3:	13
4:	18
5:	12
6:	6 → 15



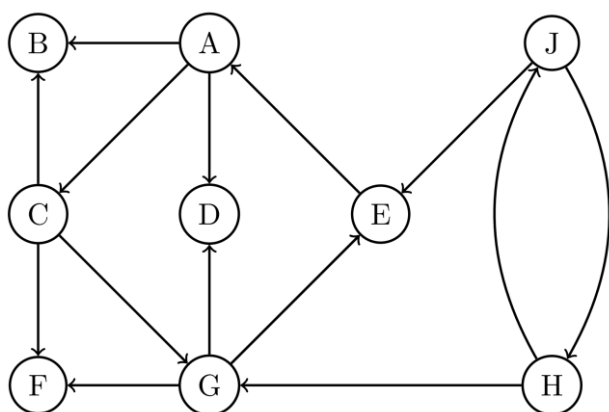
شماره دانشجویی:

بسمه تعالی
آزمون پایان ترم
ساختمان داده‌ها و الگوریتم‌ها
شنبه ۱۷ دی ۱۴۰۱
مدت آزمون ۱۲۰ دقیقه



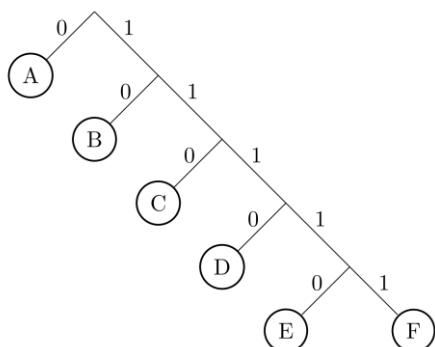
نام و نام خانوادگی:

۴ - الگوریتم جستجوی عمق اول (DFS) را می‌خواهیم بر روی گراف زیر اجرا کنیم. از گره A شروع می‌کنیم. در صورت امکان رفتن به بیش از یک گره، به گره‌ای می‌رویم که حرف آن کوچک‌تر است. در صورتی هم که تمام رئوس قابل دسترس تا کنون را تکمیل کردیم، از بین رئوس دیگر کوچک‌ترین راس را انتخاب کرده و از آن‌جا ادامه می‌دهیم. زمان شروع و خاتمه تمام رئوس (مشابه آنچه در کلاس شرح داده شد) را در جدول زیر کامل کنید. فرض کنید از زمان ۱ شروع کرده و هر بار زمان را یک واحد افزایش می‌دهیم. (۴ نمره)



حرف	زمان شروع	زمان اتمام
A	1	14
B	2	3
C	4	13
D	8	9
E	10	11
F	5	6
G	7	12
H	15	18
J	16	17

۵ - زهرا می‌خواهد یک نامه را با استفاده از کد هافمن، کدگذاری کند. در این نامه، فقط حروف {A, B, C, D, E, F} آمده است. او پس از استفاده از الگوریتم ارائه شده در کلاس برای رسم درخت کد هافمن، به درخت زیر رسیده است. اگر بدانیم که حرف‌های E و F تنها یک‌بار در نامه آمده‌اند (یعنی فرکانس تکرار آن‌ها ۱ بوده است)، آنگاه حداقل فرکانس تکرار سایر حروف چند خواهد بود؟ چرا؟ (۴ نمره)



حرف	حداقل فرکانس تکرار
A	۵
B	۳
C	۲
D	۱

فرکانس تکرار D نمی‌تواند از E و F کمتر باشد، زیرا در غیر این صورت می‌بایست با یکی از آن‌ها والد مشترک می‌داشت. در نتیجه، فرکانس تکرار D حداقل ۱ خواهد بود.

فرکانس تکرار C نباید از مجموع فرکانس تکرار E و F کمتر باشد (زیرا در غیر این صورت D به جای ادغام با E و F، با C ادغام می‌شد. پس فرکانس تکرار C هم حداقل ۱+۱ یعنی ۲ خواهد بود).

مشابه حالت قبل، فرکانس تکرار B نباید از حاصل ادغام E، D، و F کمتر باشد. در نتیجه، فرکانس تکرار B حداقل ۳ خواهد بود.

باز به صورت مشابه، فرکانس تکرار A نباید از مجموع فرکانس تکرار C، D، E، و F کمتر باشد. در نتیجه فرکانس تکرار A حداقل ۵ خواهد بود.



شماره دانشجویی:

بسمه تعالی
آزمون پایان ترم
ساختمان داده‌ها و الگوریتم‌ها
شنبه ۱۷ دی ۱۴۰۱
مدت آزمون ۱۲۰ دقیقه



نام و نام خانوادگی:

۶ - دنباله‌ای از n عدد a_1, a_2, \dots, a_n به ما داده شده است و از ما خواسته شده است که بزرگ‌ترین زیر دنباله صعودی آن را پیدا کنیم. به یک زیر دنباله صعودی می‌گوییم اگر $\{i_1, i_2, \dots, i_k\}$ زیر مجموعه‌ای از اعداد ۱ تا n باشند به گونه‌ای که $i_1 < i_2 < \dots < i_k$ و $a_{i_1} \leq a_{i_2} \leq \dots \leq a_{i_k}$.

الف) بزرگ‌ترین زیر دنباله صعودی برای دنباله زیر را پیدا کنید: (۱ نمره)

5, 3, 4, 8, 7, 10

3, 4, 8, 10

ب) فاطمه پس از کمی فکر کردن، الگوریتم زیر برای حل این مسئله به ذهنش رسید: ابتدا کوچک‌ترین عدد را به عنوان اولین عدد انتخاب کنیم. سپس اولین عدد بزرگ‌تر پس از آن را به عنوان عضو دوم، و سپس اولین عدد بزرگ‌تر از عدد دوم را به عنوان عدد سوم انتخاب کرده و این کار را این قدر ادامه دهیم تا دیگر نتوان عدد دیگری را انتخاب کرد. آیا راه حل فاطمه همواره بزرگ‌ترین زیر دنباله صعودی را به ما می‌دهد؟ اگر بله، اثبات کنید. اگر نه، یک مثال نقض بزنید. (۲ نمره)

همواره جواب بهینه را به ما نمی‌دهد. به عنوان مثال، در دنباله 1, 3, 2، این الگوریتم تنها عدد ۱ را انتخاب می‌کند، در حالیکه جواب درست زیردنباله 2, 3 است.

۷ - در کلاس مسئله کوله‌پشتی صفر و یک را تحلیل کردیم. در آن مسئله، یک کوله‌پشتی با ظرفیت W داشتیم و می‌بایست تصمیم می‌گرفتیم که کدام زیر مجموعه از n شیء را در کوله‌پشتی بگذاریم که مجموع وزن آن‌ها از W بیشتر نباشد و ارزش آن‌ها هم بیشینه باشد. حال شما می‌خواهید برای تعطیلات بین دو ترم، یک سفر تفریحی با هواپیما به جزیره کیش داشته باشید تا کمی از آلودگی و سرمای هوای تهران دور شوید. هواپیما به شما اجازه می‌دهد که دو کیف با حداکثر وزن W داشته باشید. چگونه می‌توان راه حل مسئله کوله‌پشتی را گسترش داد که این مسئله را برای شما حل کند؟ یعنی چطور می‌توان زیر مجموعه‌ای از اشیاء را در این دو کوله‌پشتی گذاشت که هر شیء حداکثر در یکی از دو کیف قرار گیرد و وزن هیچ کیف هم از W بیشتر نباشد؟ شما فقط کافی است که بیان کنید زیر مسئله بهینه به چه صورتی تعریف می‌شود، رابطه بازگشتی محاسبه ارزش بهترین حالت را بنویسید، و بیان کنید جواب مسئله اصلی کدام جواب زیر مسائل است. (۴ نمره)

زیر مسئله بهینه این است که چطور می‌توان تنها با در نظر گرفتن i شیء اول، به بهترین نحو دو کیف را پر کرد به گونه‌ای که وزن کیف اول حداکثر k و وزن کیف دوم حداکثر l باشد. اگر ارزش بهترین نحوه انتخاب را با $a[k, l, i]$ اگر وزن شیء i ام را با w_i نمایش بدهیم، آنگاه خواهیم داشت:

$$a[k, l, i] = \max \{ a[k, l, i-1], a[k-w_i, l, i-1] + v_i, a[k, l-w_i, i-1] + v_i \}$$

جواب مسئله اصلی هم $a[W, W, n]$ خواهد بود.



شماره دانشجویی:

بسمه تعالی
آزمون پایان ترم
ساختمان داده‌ها و الگوریتم‌ها
شنبه ۱۷ دی ۱۴۰۱

مدت آزمون ۱۲۰ دقیقه



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

نام و نام خانوادگی:

۸ - سجاد مسئول تقسیم کردن باقی‌مانده آجیل‌های شب یلدا بین افراد فامیل شده است. او بدون استفاده از ترازو، آجیل‌ها را به n بسته تقسیم کرد. بعداً که افراد برای تحویل گرفتن آجیل‌ها آمدند، میزان انتظار افراد برای دریافت آجیل با هم تفاوت داشت. سجاد می‌خواهد ببیند که آیا می‌تواند به گونه‌ای بسته‌ها را بین افراد تقسیم کند که هر کسی میزان آجیل دریافتی‌اش از میزان انتظارش بیشتر باشد یا نه؟ اگر تعداد افراد n نفر، میزان آجیل موجود در بسته i ام برابر با w_i و میزان آجیل انتظاری فرد i ام e_i باشد، چگونه می‌توان در زمان $O(n \lg n)$ بررسی که آیا این کار ممکن است یا نه، و اگر بله، یک جواب ممکن برای آن را پیدا کرد. (۴ نمره)

ابتدا وزن بسته‌ها و همچنین میزان انتظار افراد را مرتب می‌کنیم. برای اینکار به $O(n \lg n)$ نیاز داریم. اگر لیست مرتب‌شده وزن‌ها به ترتیب $w_1 \leq w_2 \leq \dots \leq w_n$ و لیست مرتب‌شده انتظارات $e_1 \leq e_2 \leq \dots \leq e_n$ باشد، آنگاه اینکار ممکن است اگر برای تمام i ها داشته باشیم $w_i \geq e_i$. در این حالت، کافی است بسته i ام را به فرد i ام بدهیم. واضح این که این یک جواب درست است. اما اگر برای یک i داشته باشیم $w_i < e_i$ آنگاه مسئله جواب ندارد، زیرا هیچ‌یک از بسته‌های w_1 تا w_i را نمی‌توان به افراد i تا n داد زیرا انتظار تمام آن‌ها حداقل e_i است که آن هم از تمام این بسته‌ها بزرگ‌تر است. در نتیجه اگر مسئله جواب داشته باشد، باید بسته‌های $i+1$ تا n را بین اعضای i تا n تقسیم کند که غیر ممکن خواهد بود.

۹ - رضا به شدت تغییرات قیمت روزانه دلار را دنبال می‌کند و همواره آرزو می‌کند که ای‌کاش می‌دانست قیمت دلار در روزهای آینده چگونه تغییر می‌کند تا با خرید و فروش در زمان مناسب، بتواند سودی خوبی به دست آورد. دیشب، وقتی پس از ساعت‌ها مطالعه درس ساختمان داده‌ها برای امتحان امروز خوابش برد، در یک رویای صادقه، جدولی از قیمت دلار در ماه‌های بهمن و اسفند را مشاهده کرد. او ناگهان از خواب پرید و بلافاصله قیمت‌های روزانه را روی یک ورق کاغذ نوشت. حال او فکر می‌کند که چگونه می‌تواند با استفاده از ۱۰ میلیون تومان پس‌اندازی که دارد، حداکثر سود ممکن را کسب کند. او می‌خواهد دو روز را انتخاب کند به گونه‌ای که در روز اول (مثلاً ۱۵ بهمن) تمام پس‌انداز خود را به دلار تبدیل کند و در ادامه در روز دوم (مثلاً ۱۳ اسفند) تمام دلار را بفروشد. به او کمک کنید که برای این کار یک الگوریتم طراحی کند. الگوریتم شما باید مقدار قیمت دلار در n روز را بگیرد و در زمان $O(n)$ تعیین کند بهترین روز برای خرید و سپس برای فروش چه روزی است. (۴ نمره)

ابتدا برای هر روز محاسبه می‌کنیم که حداکثر قیمت دلار در روزهای پس از آن، چقدر خواهد بود. فرض کنید که این مقدار را با $m[i]$ نمایش می‌دهیم. اگر قیمت در روز i ام برابر با $p[i]$ باشد، خواهیم داشت: $m[i] = \max \{ m[i+1], p[i+1] \}$. در یک آرایه دیگر هم نگه می‌داریم که این بیشینه در کدام روز است. حال برای هر روز، حداکثر سود حاصل از خرید در آن روز برابر با $m[i] - p[i]$ است. کافی است این مقدار را برای تمام روزها حساب کرده و روز با مقدار بیشینه را انتخاب می‌کنیم.