

امتحان پایان ترم ساختمان داده

زمان امتحان کتبی: ۱۲۰ دقیقه - قابل تمدید نیست - برای هر یک دقیقه‌ای که زودتر برگه را تحویل دهید، یک ششم نمره اضافه می‌شود.

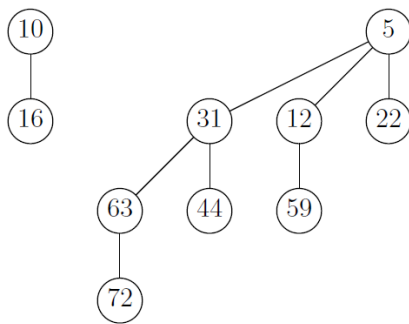
دانشجویان با شماره دانشجویی زوج، سؤالات زوج و دانشجویان با شماره دانشجویی فرد، سؤالات فرد را پاسخ دهند

تعداد سؤالات برای هر دانشجو: ۱۱ عدد -

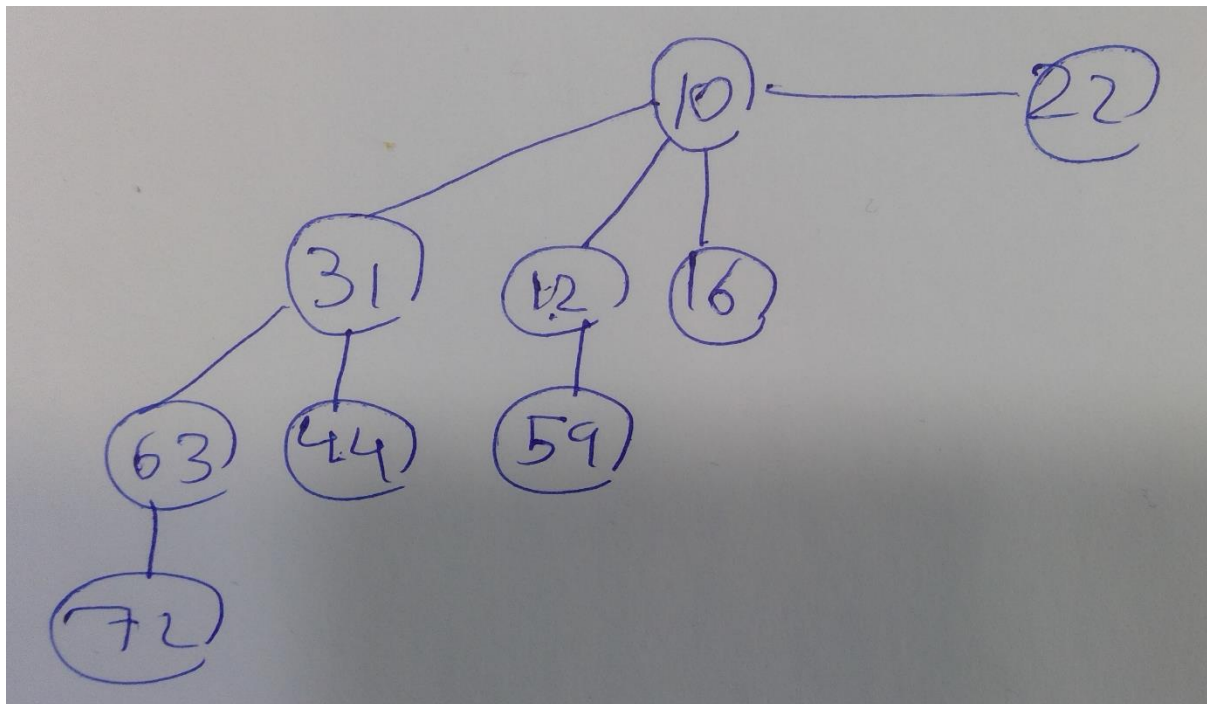
بارم همه سؤالات ۱۰ نمره است. جمعاً ۱۱۰ نمره - ۱۰ نمره بونوس

Advanced Data Structure

۱- عنصر حداقل را از Binomial-Heap روبرو حذف کنید و نتیجه را رسم کنید؟

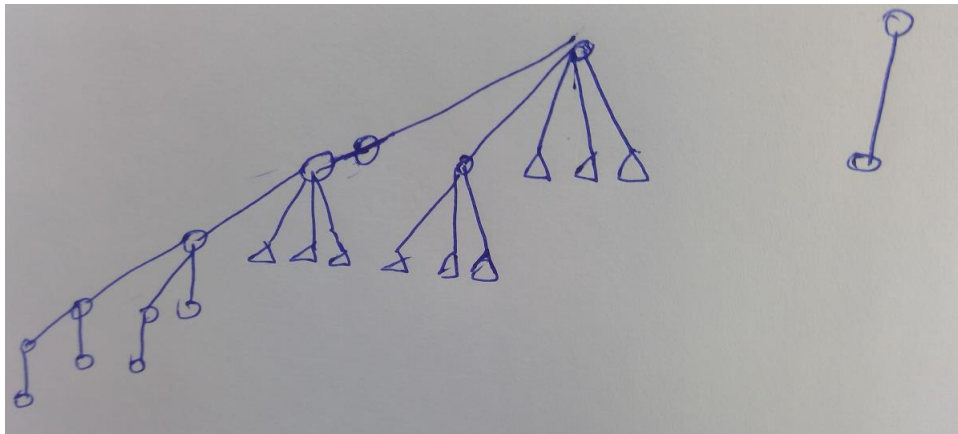


پاسخ: ۲ ریشه



۲- فرض کنید در یک Fibonacci-heap بعد از انجام عملیات مختلف درجه‌ی درخت‌های موجود در ریشه برابر 1. 1.1.1.1.3.3.3 باشد. بعد از انجام عملیات consolidate چند درخت در ریشه باقی می‌ماند؟ شکل Heap را بکشید؟

پاسخ: 2 ریشه دارد



۳- درستی یا نادرستی هر یک از گزاره‌های زیر را درباره‌ی B-tree از مرتبه‌ی m و اندازه‌ی n تعیین کنید؟ برای پاسخ خود دلیل بیاورید؟ (جمعاً ۱۰ نمره) – اگر یک گزاره درست نوشته اید، ۴ نمره – دو گزاره درست ۷ نمره –

الف) ارتفاع B-tree برابر $\log_m n$ است.

ب) هر نود حداکثر m کلید در خود نگهداری می‌کند.

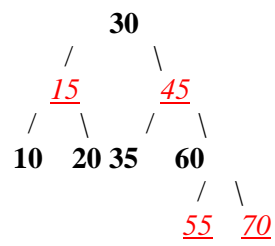
ج) B-tree در مواردی که اندازه‌ی m کم باشد (مثلاً ۵) بهتر از شرایطی که مقدار m زیاد باشد (مثلاً ۵۰۰) کار می‌کند.

پاسخ: الف درست است.

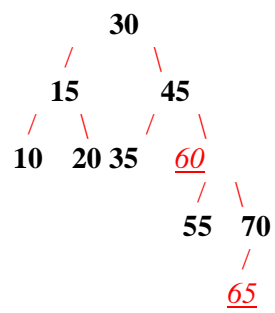
ب: نادرست است. چون حداکثر $2m-1$ عنصر دارد.

ج: نادرست، نکته برتری btree برای حالت‌هایی که m زیاد باشد و می‌توان روی دیسک با حداقل دسترسی حجم زیادی دیتا مدیریت کرد.

۴- عدد ۶۵ را در درخت قرمز-سیاه زیر درج کنید.



پاسخ:

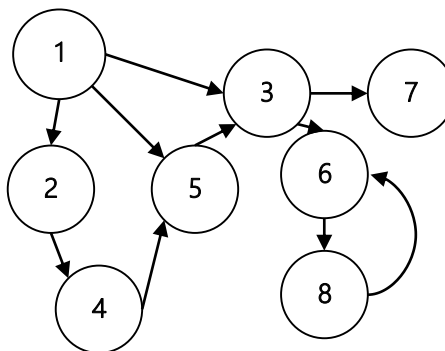


Graph

۵- الگوریتم زیر یک گراف G را به صورت DFS جستجو می کند و به رئوس شماره های DFN را نسبت می دهد. فرض کنید که گراف زیر را با $dfs(1)$ صدا کنیم و رئوس مجاور یک رأس را به ترتیب شماره های آنها مورد بررسی قرار دهیم. همچنین مقدار عددی اولیه Count را صفر در نظر بگیرید. آرایه DFN به ازای راس های ۱ تا ۸ چه مقداری است؟

```

Procedure dfs(v:Vertex)
  w:Vertex;
  Begin
    Visited[v]=true
    For all vertices  $w \in adj(v)$ 
      If (visited[w]==false) then
        dfs(w)
    Count=Count+1;
    DFN[v]=Count;
  End.
  
```



گره v	1	2	3	4	5	6	7	8
DFN(v)								

پاسخ:

گره v	1	2	3	4	5	6	7	8
DFN(v)	۸	۷	۴	۶	۵	۲	۳	۱

۶- الگوریتم DFS به روی گراف بدون جهت G اجرا شده است. زمان ورود $d(v)$ و خروج $f(v)$ از هر رأس v به صورت زیر است. کدامیک از یال های زیر در گراف G نمی تواند وجود داشته باشند؟ چرا؟ (ممکن است بیش از یک گزینه پاسخ باشد)

پاسخ درست به هر گزینه ۲,۵,۵ نمره مثبت دارد.

گره v	a	b	c	d	e	f	g	h	I
---------	---	---	---	---	---	---	---	---	---

d(v)	۱	۹	۸	۷	۵	۴	۲	۱۲	۱۳
f(v)	۱۸	10	۱۱	۱۶	۶	۱۷	۳	۱۵	۱۴

ch (۴)

bd (۳)

di (۲)

fh (۱)

جواب: صرفاً ch نمی‌تواند باشد. مابقی می‌توانند در گراف باشند.

۷- اگر $d[u]$ نخستین زمان ملاقات گره u در جستجوی عمق اول و $f[v]$ آخرین زمان ملاقات گره v و $yal(u,v)$ یک

Cross edge باشد، کدام رابطه درست است؟ (چون گزینه‌ها مانع الجمع است، فقط در صورتی که گزینه درست

(ب) انتخاب شده باشد و مابقی نادرست ذکر کرده باشید، ۱۰ نمره در غیراینصورت صفر نمره بدهید)

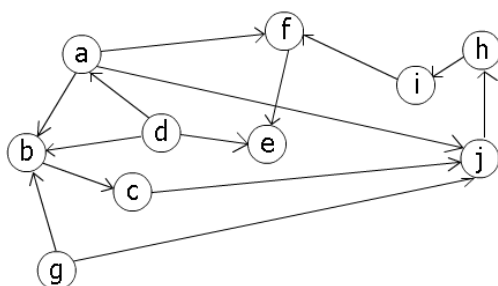
$$d[v] < f[v] < d[u] < f[u] \quad (۲) \quad d[v] < d[u] < f[u] < f[v] \quad (۱)$$

$$d[v] < d[u] < f[v] < f[u] \quad (۴) \quad d[u] < f[u] < d[v] < f[v] \quad (۳)$$

جواب: فقط گزینه ب درست است. مابقی نادرست هستند

۸- الگوریتم Topological Sort را توضیح مختصری داده و سپس روی گراف زیر پیاده‌سازی کنید. - توضیح الگوریتم ۵

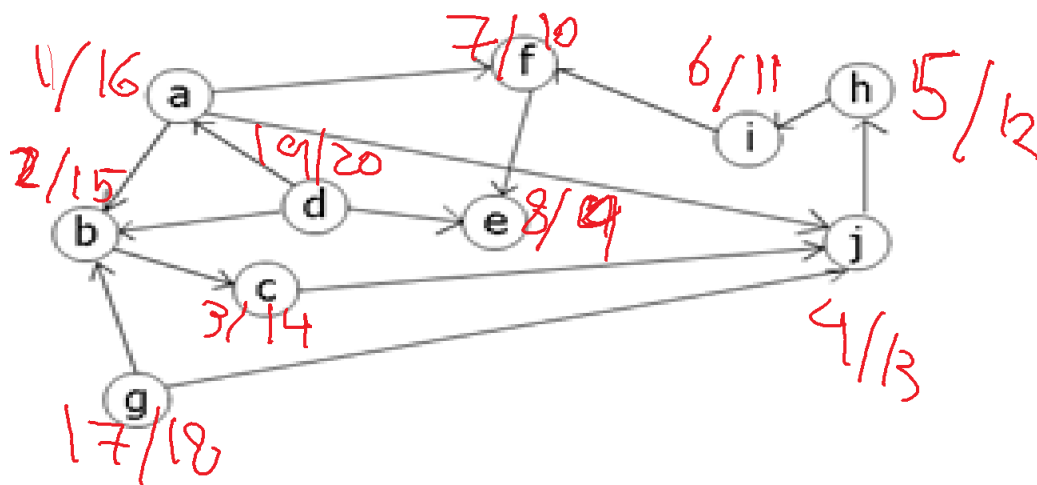
نمره - ارائه خروجی صحیح ۵ نمره



پاسخ:

یک DFS می‌زنیم و برحسب finish time به صورت نزولی گره‌ها را مرتب می‌کنیم. این ترتیب توپولوژیکی است.

عدد اول بالای گروه discovery time و عدد دوم finish time است.



ترتیب نزولی finish time

D - g - a - b - c - j - h - l - f - e

۹- فرض کنید گراف G داده شده است به صورتی که وزن یال‌های آن عددی بین صفر و N است. الگوریتمی ارائه دهید که کوتاه‌ترین مسیر از یک رأس s به t را با زمان $O(E + V \cdot N)$ و حافظه $O(V + N)$ مشخص کند.

راه حل:

یک آرایه از صف‌ها داریم، از گره s جستجوی سطح اول می‌زنیم. گره‌های مجاور را با توجه به فاصله شان در صف مربوطه به خود اضافه می‌کنیم. برای مثال اگر گره فعلی فاصله a از s داشته باشد و یال به گره مجاورش وزن b داشت، گره مجاور را در صف $a+b$ اضافه می‌کنیم. توجه کنید که صف‌ها در خود فاصله‌ها را نیز نگه می‌دارند پس لازم نیست دفعه‌ی بعدی صف‌ها با فاصله کمتر از a را بررسی کنیم. فاصله‌های نهایی گره‌ها را هنگام خارج شدن از صف مقدار دهی می‌کنیم و اگر گره‌ای را قبل از خارج شدن پیموده بودیم، دیگر در نظر نمی‌گیریم.

۱۰- گراف دو بخشی گرافی است که بتوان رؤس آن را به دودسته افراز کرد به نحوی که در بین رؤس دسته‌ها یالی وجود نداشته باشد. الگوریتمی ارائه دهید که با گرفتن اطلاعات یک گراف به ما بگوید دوبخشی است یا نه.

راه حل:

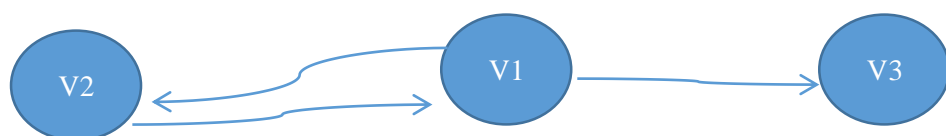
با یک بار BFS زدن روی گره‌ها حرکت می‌کنیم و گره‌ها به دو بخش تقسیم می‌کنیم. گره نخست را در یکی از بخش‌ها قرار می‌دهیم و به ترتیب گره‌هایی را که می‌بینیم در بخش مخالف قرار می‌دهیم. در حین حرکت اگر به تناقضی برخوردیم یعنی گراف دو بخشی نیست و در غیر این صورت گراف دو بخشی است زیرا به بخش مجزا افراز شده است.

۱۱- پرفسور Lauren ادعا می‌کند که الگوریتم برای اجزای همبندی قوی می‌تواند با استفاده از گراف اولیه (به جای ترانهاد) در دومین جستجوی اول عمق و بررسی رؤس به ترتیب افزایش زمان‌های خاتمه ساده‌تر شود. آیا ادعای پرفسور درست می‌باشد؟

راه حل:

نه، همواره درست نیست.

مثال نقض شکل زیر است. فرض کنید که DFS بزیم از رأس $V1$ ، سپس اگر بر اساس افزایش $FINISHING TIME$ هاشون مرتب کنیم می‌شه:



Finishing Time: V_2, V_3, V_1

حال برای دومین بار که از رأس V_2 ما DFS بزنیم. و به ترتیب افزایش Finish Time ها مرتب کنیم
Finish Time : V_3, V_1, V_2

والگوریتم صرفاً یک SCC خروجی میدهد. در حالیکه دو تا SCCs در گراف داریم. $\{v_1, v_2\}, \{v_3\}$

۱۲- اگر یک یال به گراف اضافه کنیم. اجزای همبند قوی گراف چگونه تغییر می کند؟ آیا افزایش می یابد یا کاهش؟ به چه میزان ممکن است تغییر یابد؟

راه حل:

دو حالت رخ می دهد. یا اینکه تغییر نمی کند. یا کاهش می یابد.

فرض کنید SCCs گراف اصلی $m =$

SCCs گراف جدید $m' =$

داریم: $m' \leq m$, $m' \geq 1$

ممکن است تغییرات بیش از یک باشد.

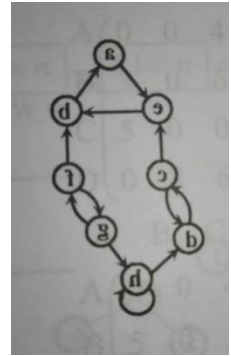
۱۳- الگوریتمی ارائه دهید تا تعداد یال هایی که باید به یک گراف اضافه کنیم. تا قوی همبند شود را محاسبه کند؟ هزینه این الگوریتم را بیابید.

راه حل:

ابتدا از گراف DFS می زنیم. و بعد از ترانهاده آن DFS می زنیم. با این کار مولفه های قوی همبند به دست می آیند.
حال یک گراف جدید ایجاد می کنیم. که رئوس آن مولفه های قوی همبند هستند. و اگر بین دو مولفه قوی همبند یال وجود داشته باشد. در گراف جدید بین آن دو یال جهت دار می گذاریم. به وضوح گراف جدید یک DAG است. (چرا که اگر دور داشت به این معنی بود که رئوس روی دور همگی به یکدیگر راه داشتند. و از ابتدا مولفه های قوی همبند جداگانه نبوده است. و باید همگی یک مولفه می شدند.)

در این DAG تعداد رئوسی که یال ورودی ندارند. را می شماریم. و آن را $C1$ می نامیم. سپس تعداد رئوسی که یال خروجی ندارند. را می شماریم. و آن را $C2$ می نامیم. اگر DAG فقط یک رأس داشته باشد. لازم نیست یالی اضافه کنیم. در غیر این صورت باید $\text{MAX}(C1, C2)$ یال اضافه کنیم.

۱۴- گراف جهت دار زیر چند مولفه متصل به صورت قوی دارد؟



راه حل:

۴ تا $\{a, e, b\}$, $\{g, f\}$, $\{c, d\}$, $\{h\}$

Amortized Analysis

۱۵- پشته‌ای با حداکثر اندازه‌ی k را در نظر بگیرید. بعد از اجرای k عمل به‌طور خودکار از محتوای پشته یک نسخه‌ی پشتیبان تولید می‌شود. هزینه‌ی اجرای n عمل با در نظر گرفتن copy به‌عنوان یک عمل از چه مرتبه‌ی زمانی است؟ دقیقترین پاسخ را بگویید؟ چرا؟

الف) $O(k^2)$ ب) $O(n)$ ج) $O(\log(n/k))$ د) $O(\log^k n)$

پاسخ: گزینه ب درست‌ترین است. هر عمل Push/pop با هزینه $O(1)$ انجام می‌گیرد. بعد از اجرای k عمل یک پشتیبان گیری داریم که قاعدتاً از $O(K)$ است (چون اندازه پشته قطعاً کمتر مساوی K است)، پس به ازای k عمل کل هزینه‌های مورد نیاز به‌صورت زیر است:

$$K * 1 + K = 2K$$

یعنی به‌صورت متوسط هر عمل هزینه ۲ دارد. یعنی $O(1)$ پس برای N عمل $O(N)$ می‌شود

۱۶- فرض می‌کنیم ساختمان داده‌ای مبتنی بر پشته ساخته‌ایم این ساختمان داده از عملی به نام $\text{MultiPush}()$ پشتیبانی می‌کند این عمل به این صورت است که K عنصر را درون پشته درج می‌کند آیا هزینه سرشکنی عملیات مربوط به پشته در این ساختمان داده نیز $O(1)$ باقی می‌ماند؟ اگر جواب شما مثبت است می‌بایست یک تابع پتانسیل مناسب ارائه کنید در غیر این صورت توجیه کنید که هزینه‌ی سرشکنی مربوط به این ساختمان داده $O(1)$ نمی‌باشد.

پاسخ:

خیر دیگر هزینه‌ی سرشکنی $O(1)$ باقی نمی‌ماند چرا که هزینه‌ی مربوط به عملیات‌های پشتیبانی شده توسط این ساختمان داده وابسته به عنصرهای درج شده می‌باشد، از آنجا که عمل هزینه‌ی $\text{MultiPush}()$ برابر با $\Theta(k)$ می‌باشد بنابراین هزینه

انجام n عمل $\text{MultiPush}()$ که هر کدام k عنصر را درج می‌کند برابر با $\Theta(nk)$ می‌باشد در نتیجه هزینه سرشکنی برابر با $\Theta(k)$ می‌باشد.

۱۷- یکی از راه‌های ساختن صف استفاده از دوپشته می‌باشد این روش را توضیح دهید و نشان دهید صفی که با استفاده از این روش ساخته می‌شود هزینه سرشکنی هر Enqueue و هر Dequeue برابر با $O(1)$ می‌باشد.

پاسخ:

برای انجام دادن عمل Enqueue می‌بایست در پشته ی اول عمل درج را انجام دهیم و برای Dequeue از پشته ی دوم عمل pop کردن را انجام می‌دهیم اگر پشته ی دوم خالی بود می‌بایست تمام عناصر پشته ی اول pop شود و درون پشته ی دوم درج شوند با این منطق هزینه هر عمل Enqueue اعتباری برابر با ۳ در نظر می‌گیریم و برای هر عمل Dequeue هزینه برابر صفر در نظر می‌گیریم چرا برای هر عنصری که Enqueue می‌شود ما اعتباری معادل با ۲ واحد ذخیره کرده‌ایم برای انجام عمل Dequeue این عنصر می‌بایست به پشته ی دوم منتقل شود و یک واحد نیز صرف این انتقال می‌شود و هزینه‌ای معادل با یک واحد برای Pop شدن این عنصر از پشته ی دوم نیاز است. بنابراین هزینه سرشکنی برابر با $O(1)$ می‌باشد.

۱۸- نشان دهید اگر در در شمارنده‌ی k بیتی عملگر کاهش هم وجود داشت، هزینه‌ی انجام n عمل متوالی برابر با $\Theta(nk)$ می‌باشد.

پاسخ: فرض کنید عدد موجود شامل یک عدد ۱ است که $k-1$ تا عدد صفر به دنبال خودش دارد. در این حالت اگر شمارنده را کاهش دهیم تمام این k بیت را به ۱ تبدیل کرده و آن ۱ را به صفر تبدیل می‌کند. حال اگر دوباره افزایش دهیم، تمام آن k بیت به صفر تبدیل شده و $k+1$ امین صفر به ۱ تبدیل می‌شود. اگر همین فرآیند را n بار تکرار کنیم، هزینه‌ی آن $\Theta(nk)$ خواهد شد.

Algorithms

۱۹- مرتبه زمانی الگوریتم‌های زیر به ترتیب از راست به چپ کدام گزینه می‌باشد؟

A. فرض کنید $X[1 \dots n]$ و $Y[1 \dots n]$ دو آرایه مرتب شده از اعداد باشند، الگوریتمی که عضو میانه در میان این $2n$ عدد را بیابد.

B. فرض کنید که لیست مرتب A را که از اعداد صحیح پر شده است داریم. الگوریتمی که نشان دهد آیا در بین این اعداد، عددی با مشخصه $A[i] = i$ وجود دارد. به‌طور مثال: $\{-1.0.0.3.7\}$ جواب true باز می‌گردد

$$A[3] = 3$$

$$O(\lg n) - O(n) \quad \text{ب)}$$

$$O(\lg n) - O(\lg n) \quad \text{الف)}$$

$$O(n) - O(n) \quad \text{د)}$$

$$O(n) - O(\lg n) \quad \text{ج)}$$

ج) گزینه الف

الگوریتم A: اگر آرایه $X = X_1 - a - X_2$ در نظر بگیریم که a عنصر وسط و X_1, X_2 به ترتیب نیمه قبل و نیمه بعد از a باشند و آرایه $Y = Y_1 - b - Y_2$ باشد. و $a < b$ باشد. بنابراین حتماً جواب در قسمت X_1 و Y_2 نیست. پس با $O(1)$ می توان نصف دو آرایه را حذف کرد. این عمل را می توان با $O(\log n)$ انجام داد.

الگوریتم B: در هر لحظه با بررسی عنصر وسط این کار عملی است.

۲۰- دو آرایه‌ی مرتب شده به طول های m و n در نظر بگیرید. می خواهیم k امین عضو اجتماع این دو آرایه را به دست آوریم. کمترین زمان اجرای این کار را به دست آورید. (بهترین گزینه را انتخاب کنید).

الف) $O(\lg m + \lg n)$

ب) $O(\sqrt{m} + \sqrt{n})$

ج) $O(m + n)$

د) $O(\lg m * \lg n)$

جواب: الف - متوسط

ایده الگوریتم بسیار شبیه binary Search است. بدین صورت که همواره عنصر وسط یکی از آرایه ها را با جستجوی دودویی در درون آرایه دوم و یافتن اندیس عدد یافت شده در آرایه دوم، تکرار می کنیم. بر حسب طول آرایه و محل یافت شده و عدد K ، به صورت بازگشتی برای بخشهای بعدی تصمیم گیری می شود. حداکثر هم $\log m + \log n$ تکرار خواهیم داشت.

Hashing

۲۱- در جدول درهم سازی (hashing) با واریسی خطی (linear probing)، اگر تابع درهم سازی برای هفت عنصر ورودی به صورت زیر باشد،

key	A	B	C	D	E	F	G
hash	۳	۵	۳	۴	۵	۶	۳

کدام یک از گزینه های زیر نمی تواند حاصل درج این عناصر با هر ترتیب دلخواه در آرایه ی ۷ تایی $H[0..6]$ (که در ابتدا تهی است) باشد؟ چرا؟ (صرفاً یک پاسخ درست است و بنابراین در صورتیکه صرفاً پاسخ درست را تشخیص داده باشید و مابقی را نادرست تشخیص داده اید، ۱۰ نمره مثبت درگیر این صورت صفر نمره دارد)

$$H[0..6] = [E F G A C B D] \quad .a$$

$$H[0..6] = [C E B G F D A] \quad .b$$

$$H[0..6] = [B D F A C E G] \quad .c$$

$$H[0..6] = [C G B A D E F] \quad .d$$

جواب: ب

۲۲- فرض کنید که یک جدول درهم سازی (hash) به اندازه ی ۸۰ (هشتاد) از روش آدرس دهی باز (open Addressing) استفاده می کند. ابتدا جدول خالی بوده است و تنها عملیات اضافه کردن و جستجو روی جدول انجام شده است. در حال

حاضر وضعیت درایه‌های ۴۵ تا ۵۶ جدول به صورت زیر است. اعداد بالای آرایه، اندیس درایه‌ها و اعداد پایین آرایه خروجی تابع درهم‌سازی است. اگر یک بار دیگر از جدول درهم‌سازی خالی شروع کنیم و همان عملیات را به غیر از اضافه کردن کلید e دوباره انجام دهیم، در درایه‌ی ۵۰ چه کلیدی قرار می‌گیرد؟ **(صرفاً یک پاسخ درست است و بنابراین**

در صورتیکه صرفاً پاسخ به گزینه درست داده‌اید، ۱۰ نمره مثبت در غیر این صورت صفر نمره)

۴۵	۴۶	۴۷	۴۸	۴۹	۵۰	۵۱	۵۲	۵۳	۵۴	۵۵	۵۶
A	b	c	d	e	f	g	h	i	j		
۴۶	۴۶	۴۶	۴۷	۴۶	۵۱	۴۷	۴۶	۴۸	۴۹		

g (۴

h (۳

f (۲

i (۱

جواب: د -