دانشکده مهندسی برق و کامپیوتر | مدت آزمون: ۱۰۰ دقیقه | ساختمان های داده و الگوریتم - پاییز ۱۴۰۰

شماره دانشجویی: | نام و نام خانوادگی:

به سوالاتی که **"باقیمانده شماره سوال به عدد ۳ برابر با باقیمانده یکان شماره دانشجویی شما به عدد ۳ است "** پاسخ دهید.

۸۱۰۱۱۱۱۱۷ ➡

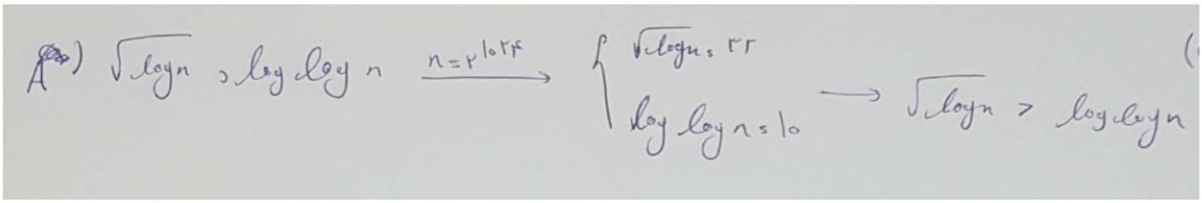
۷ ٪ ۳ = ۱ Questions: ۱, ۴, ۷, ...

۷ ٪ ۳ = ۲ Questions: ۲, ۵, ۸, ...

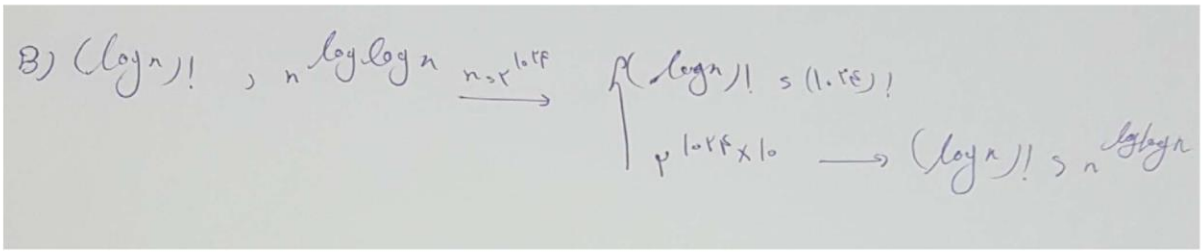
۷ ٪ ۳ = ۰ Questions: ۳, ۶, ۹, ...

توابع زیر را از نظر سرعت رشد مقایسه کنید. (۵ نمره)

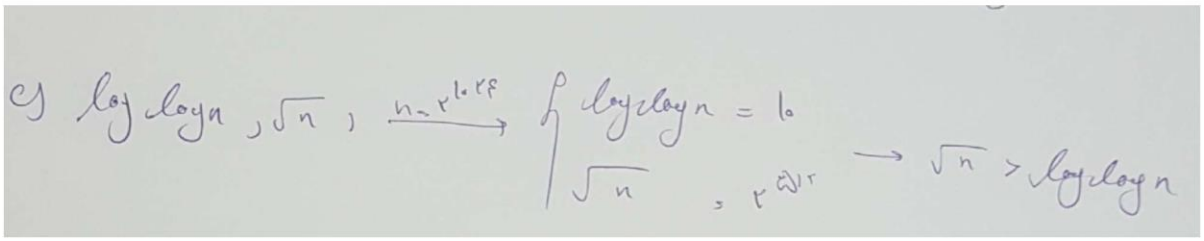
1. $\sqrt{\log n}, \log \log n$
2. $(\log n)!, n^{\log \log n}$
3. $\log \log n, \sqrt{n}$



.2



.3



۴) در عبارت محاسباتی 9 + (8 * (6+7) * 5 + 4 * (2+3)) عملگر + بر * اولویت دارد. عبارت پیشوندی آن را بکشید؟ (۵نمره)

پاسخ:

$$\left(\left(\underbrace{۲+۳}_{A=+۲۳}\right)*۴+۵*\underbrace{\left(۶+۷\right)}_{B=+۴۵}*۸\right)+۹\rightarrow\left(\underbrace{A*B*C*۸}_{***ABC۸}\right)+۹\rightarrow$$

$$\rightarrow +ABC۸۹\rightarrow +***+۲۳+۴۵+۶۷۸۹$$

۵) در عبارت محاسباتی 9 + (8 * (6+7) * 5 + 4 * (2+3)) عملگر + بر * اولویت دارد. عبارت پسوندی آن را بکشید؟ (۵نمره)

{(23+)*45+*(67+)*8}+9 = {(23+)(45+)(67+)8***}9+ = 23+45+67+8***9+

۶) در عبارت محاسباتی 9 * (8 + (6*7) + 5 * 4 + (2*3)) عملگر + بر * اولویت دارد. عبارت پسوندی آن را بکشید؟(۵نمره)

{ ((23*)4+) * ((5(67*)+)8+) } * 9 = { 23*4+567*+8+* } * 9 = 23*4+567*+8+* 9 *

پیچیدگی زمانی روابط بازگشتی زیر را به روش دلخواه محاسبه کنید. (۱۰ نمره)

7. $T(n) = 2T(n-1) + n$

8. $T(n) = T(\sqrt{n}) + O(\log(\log n))$

9. $T(n.k) = T\left(\frac{n}{2}.k\right) + T\left(n.\frac{k}{4}\right) + kn$

پاسخ ۷:

b) $T(n) = 2T(n-1) + n = 2(2T(n-2)) + n-1 + n$

$= 4T(n-2) + 2(n-1) = 8T(n-3) + 4(n-2) + 2(n-1)$

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

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

$\sum_{i=0}^{n-2} 2^i (n-i) = n\sum_{i=0}^{n-2} 2^i - \sum_{i=0}^{n-2} i2^i = n(2^{n-1}-1) - \frac{2^n n - 3.2^n + 4}{2}$

$= n(2^{n-1}-1) - (n2^{n-1} - 3.2^n + 2) = 3.2^{n-1} - n - 2$

$\Rightarrow T(n) = 2^{n-1} + 3.2^{n-1} - n - 2 = 2^{n+1} - n - 2 = \boxed{O(2^n)}$

پاسخ ۸:

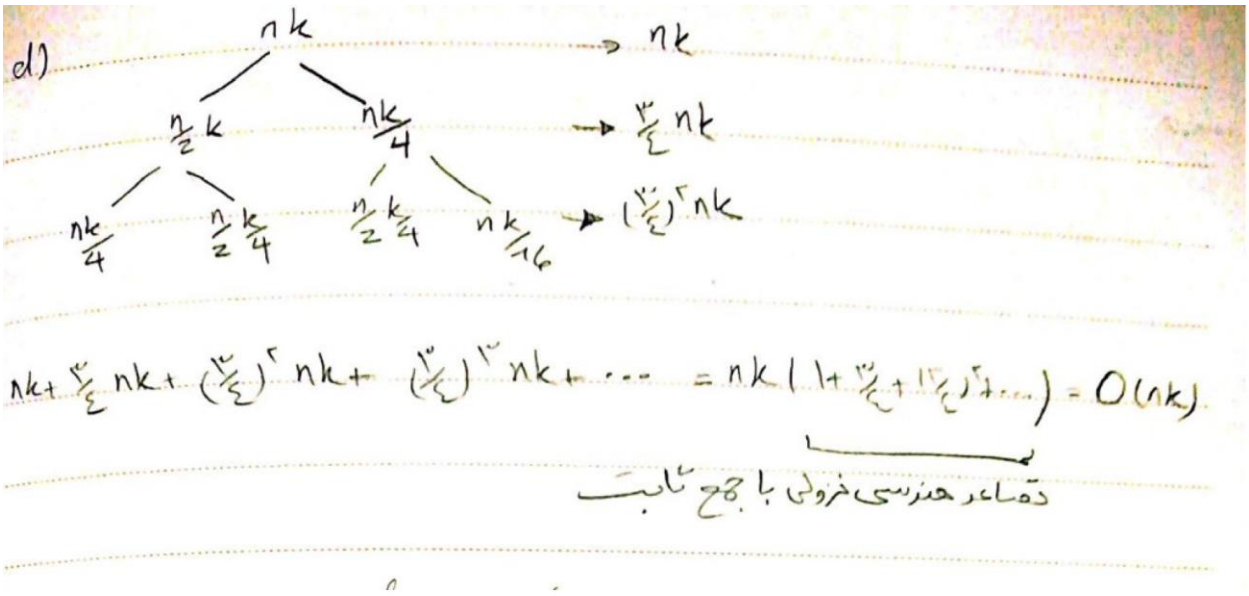
تغییر متغیر : $n = 2^{2^m} \rightarrow T(2^{2^m}) = T(2^{2^{m-1}}) + O(m)$

تغییر تابع : $F(m) = F(m-1) + O(m)$

جایگذاری : $F(m) = O(m^2)$

$\Rightarrow T(n) = O((\log \log n)^2)$

پاسخ ۹:



مرتبه زمانی قطعه کد های زیر را بدست آورید. (۵ نمره)

10.

```
for (int i = 0; i < n + 100; ++i) {
      for (int j = 0; j < i * n ; ++j){
            sum = sum + j;
      }
      for (int k = 0; k < n + n + n; ++k){
            c[k] = c[k] + sum;
      }
}
```

پاسخ: O(N^3)

11.

```
for (int j = 4; j < n; j=j+2) {
      val = 0;
      for (int i = 0; i < j; ++i) {
            val = val + i * j;
            for (int k = 0; k < n; ++k){
                  Val++;
            }
      }
}
```

پاسخ: O(N^3)

12.

```
for (int i = 0; i < n * 1000; ++i) {
      sum = (sum * sum)/(n * i);
      for (int j = 0; j < i; ++j) {
            sum += j * i;
      }
}
```

پاسخ: O(N^2)

در سوالات زیر، ضمن اشاره به هزینه زمانی، مختصر توضیح دهید. (5 نمره)

13) بدترین هزینه زمانی $T(n)$ برای insert کردن n عدد به یک درخت binary search خالی کدام است؟

پاسخ: T(n) = O(nlgn n)

14) بدترین هزینه زمانی $T(n)$ برای pop کردن یک مقدار از یک binary heap از سایز n کدام است؟

پاسخ: T(n) = O(lgn n)

15) بدترین هزینه زمانی $T(n)$ برای insert کردن یک مقدار از یک binary heap خالی کدام است؟
پاسخ: T(n) = O(nlgn n)

16) اعداد ۱ , ۲ , ... n به ترتیب از راست وارد یک پشته میشوند. در هر زمان می توانیم از پشته یک شماره بیرون آوریم و در پایانه بنویسیم. نشان دهید که میتوان به عنوان خروجی به جایگشت P1, P2, P3, ... Pn رسید اگر و تنها اگر هیچ سه i, j, k پیدا نشود که Pj < Pk < Pi , i < j < k.
۱۰ نمره

اگر $j < k$ و $P_j < P_k$، باید P_j را پیش از زمانی که P_k را در پشته قرار می‌دهیم، بیرون آوریم. اگر $P_j > P_k$ باید P_k را در پشته باقی بگذاریم تا P_j نیز در پشته قرار گیرد. این دو رخداد $i < j < k$ و $P_j < P_k < P_i$ را ناشدنی می‌سازد، چرا که P_j بایستی پیش از P_k بیرون آورده شود و پس از P_i. در حالیکه P_i پس از P_k ظاهر می‌شود.

از طرف دیگر، یک جایگشت می‌تواند با الگوریتم زیر به دست آید:

برای $j = ۱, ۲, ۳, ..., n$، صفر یا بیشتر شماره در پشته قرار دهید تا P_j برای نخستین بار در استک قرار گیرد. سپس P_j را بیرون آورده در پایانه می‌نویسیم. الگوریتم شکست می‌خورد تنها اگر به jای برسیم که P_j در سر پشته قرار نداشته باشد اما با برخی شماره‌های P_k پوشیده شده باید برای $k > j$. از آنجایی که ارزش شماره‌ها در پشته یک‌نواخت افزایش می‌یابد $P_j < P_k$. همینطور که P_k باید در پشته قرار گرفته باشد چون از یک P_i کمتر بوده است که $i < j$.

17) با استفاده از دو صف، دو روش برای پیاده سازی استک ارائه دهید. پیچیدگی زمانی عملیات push و pop را در هر یک از دو روش محاسبه کنید.
۱۰ نمره

روش اول:

Push(x): x را در صف اول قرار می‌دهد.

Pop(x): همه عناصر صف اول را تا زمانی که تنها یک عنصر باقی بماند، به ترتیب خارج می‌کند و در صف دوم قرار می‌دهد. آخرین عنصر صف اول را خارج می‌کند و در یک متغیر نگه‌داری می‌کند. سپس یا جای صف اول و دوم را عوض می‌کند و یا تمامی عناصر را از صف دوم به صف اول بر می‌گرداند. در نهایت عنصر ذخیره شده را باز می‌گرداند.

عملیات push مرتبه O(1) و pop مرتبه O(n) است.

روش دوم:

Push(x): اگر صف اول خالیست x را در آن قرار می‌دهد. در غیر این صورت x را در صف دوم قرار داده و سپس تمامی عناصر صف اول را به ترتیب خارج کرده و در صف دوم قرار می‌دهد. در نهایت تمامی عناصر صف دوم به ترتیب خارج شده و در صف اول قرار می‌گیرد.

Pop(x): اولین عنصر صف اول را خارج کرده و بر می‌گرداند.

عملیات push مرتبه O(n) و pop مرتبه O(1) است.

18) الگوریتمی با پیچیدگی زمانی O(nlogn) ارائه دهید که بزرگترین عدد ساخته شده از اعداد داخل آرایه‌ای (بطور n) شامل اعداد ۰ تا ۹ را خروجی دهد به طوری که هر عدد حداکثر یک بار تکرار شده باشد و بر اعداد ۲، ۳ و ۵ بخش پذیر باشد.(اگر در آرایه عدد ۳، دو بار آمده باشد در عدد خروجی رقم ۳ می‌تواند دو بار تکرار شده باشد.)

۱۰ نمره

جواب:

برای آن که هم بر ۲ و هم بر ۵ بخش پذیر باشد، حتما باید یکان عدد صفر باشد پس اگر آرایه شامل صفر نباشد، اعلام می‌کنیم نمی‌توان با آرایه ورودی شرط مسئله را برآورد کرد. حال بزرگترین عدد بخش پذیر بر ۳ را به دست می‌آوریم. برای این کار عناصر آرایه را به صورت صعودی مرتب می‌کنیم. اعداد می‌توانند بر ۳ باقی‌مانده ۰،۱ و ۲ داشته باشند پس برای هر کدام صف‌هایی با نام queue0 , queue1, queue2 در نظر می‌گیریم و روی آرایه مرتب شده پیمایش می‌کنیم و اعداد را در صف مناسبشان enqueue می‌کنیم. سپس مجموع تمام ارقام آرایه را به دست می‌آوریم. ۳ حالت ممکن است:

- بر ۳ بخش پذیر است. تمام اعضای ۳ صف را از آن‌ها خارج کرده و به صورت نزولی مرتب می‌کنیم و به عنوان خروجی می‌دهیم.
- باقی‌مانده بر ۳ برابر ۱ باشد. در این حالت اگر queue1 حداقل یک عضو داشته، عضو اول آن را dequeue می‌کنیم در غیر این صورت اگر queue2 حداقل ۲ عضو داشت، دو عضو از آن dequeue می‌کنیم. اگر هیچ کدام از شرایط گفته شده برقرار نبود، در خروجی اعلام می‌کنیم مسئله را پاسخ ندارد.
- باقی‌ماندا بر ۳ برابر ۲ باشد. در این حالت اگر queue2 حداقل یک عضو داشت، عضو اول آن را dequeue می‌کنیم در غیر این صورت اگر queue1 حداقل ۲ عضو داشت، دو عضو از آن dequeue می‌کنیم. اگر هیچ کدام از شرایط نبود، در خروجی اعلام میکنیم مسئله پاسخ ندارد.

در نهایت تمام اعضای صف‌ها را در یک آرایه موقتی dequeue می‌کنیم و به صورت نزولی مرتب می‌کنیم و به عنوان خروجی برمی‌گردانیم.

19) فرض کنید در ابتدا هیچ عددی در حافظه نداریم. در جریان ورودی هر بار یک عدد وارد میشود پس از insert آن در حافظه می خواهیم میانه همه اعداد موجود در حافظه شامل عدد insert شده را با کمترین هزینه برگردانیم. الگوریتمی ارائه دهید که این کار را انجام دهد.
15 نمره

جواب ۲: ایده آن است که یک درخت min-heap برای اعداد بزرگتر از میانه و یک درخت max-heap برای اعداد کوچکتر از میانه در نظر بگیریم و تعداد عناصر موجود در آن را هربار چک کنیم و اگر بالانسش بهم خورد از یک درخت پاپ و به درخت دیگر پوش کنیم و میانه را جا به جا کنیم

Time: O(logn)

20) درخت دودویی A دارای n گره و درخت دودویی B دارای m گره است که هیچ عنصر تکراری در آن ها نیست. الگوریتمی از $O(m + n)$ طراحی کنید که نشان دهد این دو درخت یکسان هستند یا خیر؟ درستی الگوریتم را توضیح دهید.
15 نمره

راه اول: الگوریتم زیر را به صورت بازگشتی به روی هر یک از گره‌های یکسان درخت و با شروع از ریشه درخت‌ها می‌زنیم:

۱. اگر هر دو ریشه پوچ بودند، مقدار ۱ باز می گردانیم.

۲. اگر پوچ نبودند، سه چیز را چک می کنیم:

۱.۲. آیا ریشه‌ها مقادیر یکسانی دارند.

۲.۲. آیا زیر درخت‌های سمت چپ هر دو ریشه با هم برابرند. (بازگشتی)

۳.۲. آیا زیر درخت‌هااای سمت راست هر دو ریشه با هم برابرند. (بازگشتی)

۳. اگر ۳ شرط مرحله ۲ درست بود ۱ بر می گردانیم و در غیر این صورت صفر.

در این حالت تمام گره‌های هر دو درخت باید چک شود. بنابراین پیچیدگی آن $O(n+m)$ می‌شود.

راه دوم: آخرین عنصر در پس ترتیب همان ریشه در درخت است. بنابراین در میانترتیب می چرخیم و ریشه را پیدا می کنیم و میان‌ترتیب را به دو زیر درخت چپ و راست تقسیم می کنیم که هر یک از آن ها نیز پیمایش میانترتیب است. حال عنصر بعدی در نمایش پس ترتیب، ریشه زیر درخت سمت راست است و در نمایش میان ترتیب نیز آن را، پیمایش سمت راست ریشه را به دو بخش تقسیم می کند. همین کار را آن‌قدر ادامه می‌دهیم تا زیر درخت سمت راست به طور کامل ساخته شود (یعنی هیچ عنصری در سمت راست ریشه درخت اصلی در نمایش میانترتیب نمانده باشد که مکان آن در زیر درخت سمت راست ریشه درخت اصلی مشخص نشده باشد). پس از این، همین کار را بر روی زیر درخت ایجاد شده در سمت چپ ریشه درخت اصلی انجام می‌دهیم. عنصری که در پیمایش پس ترتیب پس از آخرین عنصر زیر درخت سمت راست می آید، ریشه زیر درخت سمت چپ ریشه اصلی درخت خواهد بود. این کار را تا زمانی انجام می‌دهیم تا جایگاه تمام عناصر در پیمایش پس ترتیب در درخت اصلی مشخص شود. طبق این الگوریتم همواره زیر درخت‌های سمت چپ و راست در تمام عناصر یکتا هستند و می توان گفت که درخت مورد نظر یکتاست.

بنابراین کافیست پیمایش پس ترتیب و میان ترتیب دو درخت یکسان باشد. از آن‌جا که نوشتن زیر پیمایش‌ها و چک کردن آن‌ها پیچیدگی $O(n)$ و $O(m)$ دارد بنابراین پیچیدگی کلی الگوریتم $O(m+n)$ است.

21) درختی با n گره w, v پایین ترین گرهی است که هر دو این گره ها جز فرزندان یا نوادگان باشند (هر گره جزو نوادگان خودش است) الگوریتمی بهینه برای پیدا کردن جد مشترک دو گره w, v ارائه دهید و مرتبه زمانی آن را تعیین کنید.
15 نمره

میدانیم که پدر گره i گره $\lfloor i/2 \rfloor$است . تا هنگامی که این مقدار بزرگتر از یک است این عمل را تکرار می کنیم و در یک آرایه دیگر ، مقدار خانه های ارایه را که با تقسیم vبر ۲ بدست می اید trueمی کنیم برای مثال مقادیر $\lfloor \lfloor v/2 \rfloor /2 \rfloor$ $\lfloor v/2 \rfloor$ را که عملا اجداد v هستند ، در آرایه جدید trueشده اند . از آن جایی که درخت nگره دارد تعداد اجدادش از O (lgn)است حال سراغ wمی رویم و اجداد آن را بررسی می کنیم اولین جایی که اندیس آن از اجداد w در آرایه trueبود آن راس کوچکترین جد مشترک v,wاست که تعداد اجداد w نیز o(lgn) است پس در مجموع O(lgn) است.

22) آرایه ای از 5n عدد صحیح از ۱ تا n داریم. الگوریتمی با مرتبه زمانی خطی ارائه دهید که به کمک آن بتوان اعداد تکراری را حذف کرد.
15 نمره

چون تمامی اعداد صحیح بوده و در بازه 1 تا n میباشند میتوان از count sort برای مرتب کردنشان استفاده کرد.بنابراین هزینه O(5n + n) = O(n) را داده و اعداد را مرتب میکنیم.حال از یک استک برای حذف اعداد تکراری استفاده میکنیم.بدین گونه که عدد اول را در استک push میکنیم.حال برای عناصر بعدی،اگر عنصر جاری با عنصر سر استک یکی بود،سراغ عنصر بعدی می رویم و درغیر اینصورت آنرا در استک push میکنیم و همین کار را برای عناصر بعدی تکرار میکنیم.در آخر،اعداد موجود در استک،جواب مورد نظر ماست.

23) لیستی از 1 - n عدد صحیح که هر عدد بین ۱ تا n است، داده شده است و هیچ دو عددی برابر نیستند. الگوریتم بهینه ای ارائه دهید که به کمک آن، عددی را که در لیست نیامده بتوان پیدا کرد.
15 نمره

چون تمامی اعداد صحیح بوده و در بازه **1** تا **n** میباشند پس با هزینه **O(n)** وبا استفاده از **count sort** میتوان این اعداد را مرتب کرد.حال از ابتدای لیست مرتب شده شروع به پیمایش میکنیم.ایندکس اولین خانه ای که در آن ایندکس مربوط به آن خانه با مقدار خانه متفاوت باشد،جواب مورد نظر است.(با فرض اینکه ایندکس خانه ها از **1** تا **n** میباشد.).پس با هزینه زمانی **O(n)** عدد موردنظر یافت میشود.

24) دو هرم بیشینه که هر دو دارای n عنصر هستند، داریم که میخواهیم این دو را با هم ادغام کنیم و یک هرم با کمینه به اندازه 2n ایجاد کنیم. سریعترین الگوریتم برای انجام اینکار دارای چه مرتبه زمانی میباشد و آن را توضیح دهید.
15 نمره

سریع‌ترین الگوریتم از مرتبه‌ی $O(\log n)$ می‌باشد. بدین صورت که دو ریشه‌ی این هرم بیشینه با هم مقایسه شده و عنصر بزرگ‌تر به عنوان ریشه‌ی هرم بیشینه جدید انتخاب می‌شود و روی هرمی که این عنصر ریشه در آن قرار داشته عمل Extract-Max رو انجام می‌دهیم و سپس دو هرم جدید را به عنوان فرزند چپ و راست ریشه قرار می‌دهیم که در این حالت هرم بیشینه‌ی جدید حاصل می‌شود.

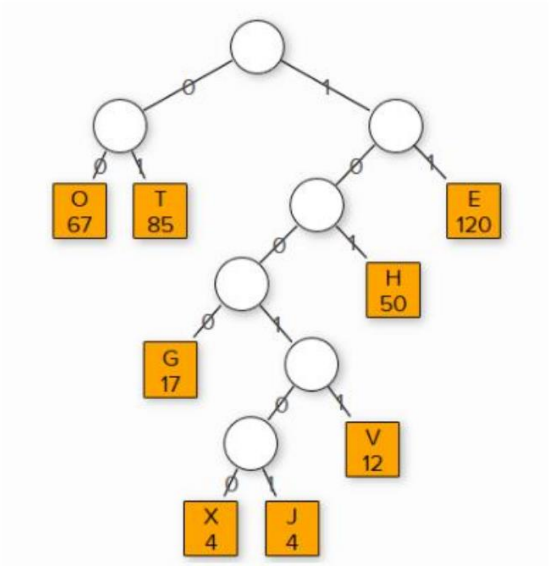
25) الف) اعداد 11, 7, 3, 8, 10, 5, 9 به یک درخت Red-Black اضافه کنید.(از راست با ذکر مراحل)
ب) عدد 10 را حذف کنید.(با ذکر مراحل)
پ) عدد 15 را اضافه کنید.(با ذکر مراحل)
(۱۵ نمره)

پاسخ: مشابه کوئیز و تمرین

26) الف) اعداد 3,8,11,10,1,5,12 به یک درخت Red-Black اضافه کنید.(از راست با ذکر مراحل)
ب) عدد 10 را حذف کنید.(با ذکر مراحل)
پ) عدد 15 را اضافه کنید.(با ذکر مراحل)
(۱۵ نمره)
پاسخ: مشابه کوئیز و تمرین

27) الف) اعداد 8,3,10,2,7,5,11 به یک درخت Red-Black اضافه کنید.(از راست با ذکر مراحل)
ب) عدد 10 را حذف کنید.(با ذکر مراحل)
پ) عدد 15 را اضافه کنید.(با ذکر مراحل)
(۱۵ نمره)
پاسخ: مشابه کوئیز و تمرین

درخت هافمن زیر را در نظر بگیرید. (5 نمره)



28) برای نمایش JTE از چه رشته ای باید استفاده کنیم؟
1001010111

29) برای نمایش THE از چه رشته ای باید استفاده کنیم؟

0110111

30) برای نمایش HGV از چه رشته ای باید استفاده کنیم؟

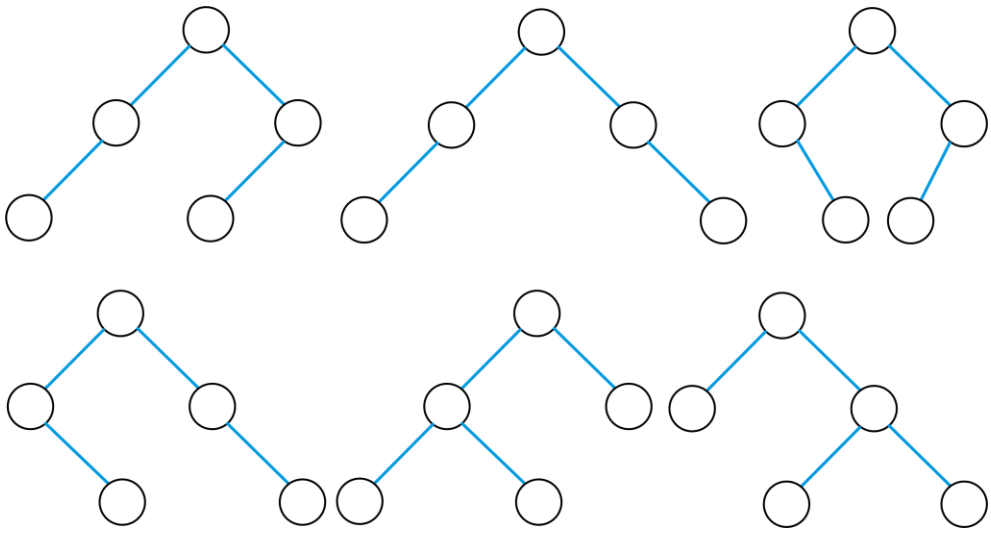
101100010011 .

31) با عناصر 1، 2، 3، 4، 5 حداکثر چندتا درخت AVL می‌توان ساخت؟ (ارتفاع درخت تهی 1- فرض می‌شود). (۵نمره)

32) با عناصر 13 ,10, 25,65,74 حداکثر چندتا درخت AVL می‌توان ساخت؟ (ارتفاع درخت تهی 1- فرض می‌شود). (۵نمره)

33) با عناصر 14,6,7,32,-1- حداکثر چندتا درخت AVL می‌توان ساخت؟ (ارتفاع درخت تهی 1- فرض می‌شود). (۵نمره)

پاسخ ۳۱-۳۲-۳۳ – هر درخت جستجوی دودویی ساخته شده با تعدادی از عناصر مشخص، دارای شکل خاص خود است. به عبارت دیگر، در درخت جستجوی دودویی، برای هر شکل درخت، تنها یک حالت عددگذاری عناصر داده شده را می‌توان یافت. بنابراین، کافی است تعداد شکل‌های درخت‌های دودویی با ۵ گره که در آن‌ها اختلاف ارتفاع زیردرخت چپ و راست همه‌ی عناصر حداکثر برابر ۱ است محاسبه شود. این شکل‌ها به صورت زیر هستند:



بنابراین، با ۵ عنصر متمایز می‌توان ٦ درخت ای‌وی‌ال ساخت.

۳٤) با داشتن پیمایش *inorder* (میان‌ترتیب) یک درخت جستجوی دودویی (*BST*) همواره می‌توان درخت را منحصر به فرد رسم کرد. (۵ نمره)

این گزاره غلط است. مثال نقض زیاد دارد.

۳۵) با داشتن پیمایش *preorder* (پیش‌ترتیب) یک درخت جستجوی دودویی (*BST*) همواره می‌توان درخت را منحصر به فرد رسم کرد. (۵ نمره)

این گزاره درست است. میتوان عناصر را مرتب کرد (در ان صورت معادل میانترتیب *BST* است) حال دو پیمایش پیش ترتیب و میان ترتیب داریم که قاعدتا یکتا میشود

۳۶) با داشـــتن پیمایش‌های *postorder*(پس ترتیب) و *preorder* (پیش‌ترتیب) یک درخت دودویی همواره می‌توان درخت را منحصر به فرد رسم کرد. (۵ نمره)

این گزاره غلط است. مثال یک درخت با دو گره است.

موفق باشید

** توجه: در صورت کشف هرگونه تقلبی طبق قوانین دانشگاه با افراد برخورد خواهد شد.