



## دانشکده مهندسی کامپیوتر

درس ساختمان‌های داده

---

تمرین سری سوم

---

مدرس ..... دکتر حسین رحمانی

طراحان ..... محمد حسین عباسپور - محمد اصولیان

تاریخ انتشار ..... ۱۴۰۱ / ۰۸ / ۲۲

تاریخ تحویل ..... ۱۴۰۱ / ۰۸ / ۲۸

### در رابطه با تمرین

➤ این تمرین شامل مباحث:

- Tree
- Heap
- BST

می باشد.

➤ برای پاسخگویی به سوالات این تمرین نیاز است که به مطالب اسلایدهای

11\_Trees(HR\_2021)\_finalized و 12\_Heap(HR\_2021)\_finalized و

13\_BinarySearchTree(HR\_2021)\_finalized مسلط باشید.

➤ نمره‌ی این تمرین از ۱۰۰ می‌باشد و بارم هر سوال روبه‌روی آن نوشته شده است.

➤ این تمرین دارای ۳ سوال عملی می‌باشد که باید با استفاده از زبان سی‌شارپ به آنها داخل سامانه

کوئرا پاسخ دهید.

➤ اگر برای جواب دادن به سوالی نیاز به پیش‌فرضی دارید، فرض خود را صریحاً نوشته و با توجه به فرض

خود به ارائه جواب بپردازید.

➤ فایل پاسخ تئوری خود را به صورت hw3\_student-id.pdf نام‌گذاری کرده و ارسال کنید.

(برای مثال hw3\_12345678.pdf)

➤ به هیچ وجه تمرینی را از دیگران کپی نکنید. در صورت مشاهده تقلب و کپی در تمرینات، نمره هر دو

طرف صفر در نظر گرفته می‌شود.

## بخش تئوری

(۱) فرض کنید اعداد بین ۱ تا ۱۰۰۰ را در یک BST داریم و می‌خواهیم برای عدد ۳۶۳ جستجو کنیم. کدام یک از دنباله‌های زیر نمی‌توانند دنباله‌ی کلیدهای جستجو شده باشند؟ چرا؟  
(۱۲ نمره)

- i. 2, 252, 401, 398, 330, 397, 363
- ii. 924, 220, 991, 244, 898, 258, 362, 363
- iii. 925, 202, 911, 240, 912, 245, 363 ✗
- iv. 2, 399, 387, 219, 266, 382, 381, 278, 363 ✓
- v. 935, 278, 347, 621, 299, 392, 356, 363 ✗

(۲) درست یا غلط بودن هر مورد را با ذکر دلیل اثبات کنید. (۱۵ نمره)

- i. همواره می‌توان بزرگترین عضو داخل یک min-heap را در مرتبه‌ی زمانی  $O(\log n)$  پیدا کرد.
- ii. داخل یک heap با ارتفاع  $d$  حداقل  $2^d$  عضو وجود دارد. (فرض کنید عمق root صفر است)
- iii. زمان اضافه کردن یک عضو داخل BST همواره از مرتبه‌ی  $O(\log n)$  می‌باشد.

(۳) آیا عملیات delete داخل BST جابه‌جایی‌پذیر است؟ یعنی آیا حذف گره‌ی  $x$  و سپس حذف گره‌ی  $y$  همان درختی را می‌سازد که حذف  $y$  و سپس حذف گره‌ی  $x$  می‌سازد؟ اثبات کنید که چرا و یا یک مثال نقض بیاورید. (۱۰ نمره)

۴) شما به همراه  $n$  نفر دیگر به یک مهمانی دعوت شده‌اید. هر فرد  $i$  در زمان  $S_i$  به مهمانی می‌رسد و در زمان  $t_i$  مهمانی را ترک می‌کند ( $S_i < t_i$ ). زمانی که یک فرد مهمانی را ترک می‌کند دیگر به مهمانی بر نمی‌گردد.

هر فرد یک معیار  $coolness$  دارد که برای فرد  $i$  مقدار آن برابر  $C_i$  می‌باشد. در مدت زمان مهمانی شما همواره با فردی که بیشترین مقدار  $coolness$  را دارد و در مهمانی حاضر است صحبت می‌کنید. (تمام مقادیر  $C$  برای افراد متمایز است)

اگر شما در حال صحبت با یک فرد باشید و فردی با  $C$  بیشتر از شخص در حال گفتگو با شما، به مهمانی بیاید؛ شما مکالمه را با فرد قبلی رها کرده و به گفتگو با شخص جدید می‌پردازید. اگر فردی که شما با آن در حال گفتگو هستید مهمانی را ترک کند، شما به سراغ مکالمه با فردی که  $C$  بیشتری از بین افراد حاضر باقی مانده در مهمانی دارد می‌روید. (ممکن است با آن قبلاً صحبت کرده باشید یا نکرده باشید).

شما اولین نفری هستید که به مهمانی می‌رسد و همچنین آخرین فردی که مهمانی را ترک می‌کند. علاوه بر آن شما محبوب‌ترین فرد هستید پس برای همین همه‌ی افراد می‌خواهند با شما صحبت کنند.

a. داده‌ساختاری را ارائه دهید که در مرتبه زمانی  $O(1)$  فردی که در هر لحظه می‌خواهید با آن صحبت کنید را به شما معرفی کند. شما همچنین باید بتوانید که این داده ساختار را در هر لحظه‌ای که یک فرد به مهمانی می‌آید یا آن را ترک می‌کند در مرتبه زمانی  $O(\log n)$  آپدیت کنید. (۱۲ نمره)

b. فرض کنید شما زمان ورودی و خروج هر شخص را می‌دانید، روشی را ارائه دهید که در مرتبه زمانی  $O(n \log n)$ ، مدت زمان کلی‌ای که شما با بقیه‌ی افراد صحبت می‌کنید را محاسبه کند. (۱۱ نمره)

دقت کنید که در بخش‌های فوق در جواب خود مرتبه زمانی را هم اثبات کنید

## بخش عملی

### • عبارت‌های ریاضی (۱۰ نمره)

هدف این سوال کار با درخت عبارت و آشنایی با شیوه‌های نمایش عبارات ریاضی است. سه حالت نمایش عبارات ریاضی infix و postfix و prefix هستند. می‌توانید از طریق [این لینک](#) یا منابع دیگر با آنها آشنا شوید. با انجام سه نوع پیمایش preorder و inorder و postorder روی درخت عبارت، می‌توانید به ترتیب حالت‌های نمایش prefix و infix و postfix را برای عبارات ریاضی به دست آورید. می‌توانید این را با درخت عبارت دلخواه روی کاغذ امتحان کنید. در این سوال به شما یک درخت عبارت داده می‌شود و از شما خواسته شده تا با انجام این سه نوع پیمایش روی درخت داده شده سه حالت نمایش عبارات ریاضی را چاپ کنید. سپس به کمک عبارت postfix و با استفاده از داده‌ساختار Stack حاصل عبارت را با دو رقم اعشار محاسبه کنید.

▪ استفاده از کتابخانه آماده برای stack مجاز می‌باشد.

### ورودی

در ورودی به شما درخت عبارت داده می‌شود. در خط اول تعداد راس‌های درخت (V) مشخص شده. تضمین می‌شود که درخت داده شده در ورودی، معتبر است. بنابراین نیازی به ذکر تعداد یال‌ها نیست. هر راس از درخت برای متمایز بودن یک id دارد که از 0 تا V-1 است. و همچنین یک مقدار دارد که شامل عملگرها (operators) و عملوندها (operands) است. در V خط بعدی راس‌های گراف مشخص شده. ابتدا id راس و سپس مقدار راس در هر خط آمده. در خط بعد id بالاترین راس (root) مشخص شده است.

در V-1 خط بعدی یال‌های درخت مشخص شده. در هر خط id دو راسی که بین آنها یال وجود دارد آمده است.

برای represent کردن یک درخت باینری می‌توانید از [این لینک](#) راهنمایی بگیرید.

```
5
0 8
1 +
2 2
3 *
4 2
1
1 0
1 3
3 2
3 4
```

### خروجی

در خروجی لازم است ابتدا پیمایش postfix سپس infix و سپس prefix را چاپ کنید.  
در آخر هم به کمک استک و postfix مقدار عبارت را محاسبه کرده و تا دو رقم اعشار چاپ کنید.

```
822*+
8+2*2
+8*22
12.00
```

### ▪ نکات مهم

- مجموعه operand ها مجموعه  $\{0,1,2,3,4,5,6,7,8,9\}$  است. از اعداد منفی و دو رقمی و اعشاری استفاده نمی‌شود.
- مجموعه operator ها مجموعه  $\{^, /, *, -, +\}$  است.

## ● درخت BST (۱۵ نمره)

هدف این سوال کار با درخت BST و توابع مربوط به آن است. برای حل این سوال لازم است که داده ساختار BST را پیاده سازی کنید. برای راحتی کارتان، پیشنهاد می شود برای کلاسی که پیاده سازی می کنید، متدهای delete و insert و search را قرار دهید. برای پیاده سازی این متدها به نکات آخر سوال حتما توجه کنید. به شما پیمایش postorder یک درخت BST داده می شود و از شما خواسته شده روی این درخت عملیاتی انجام دهید و نتیجه را برگردانید. اولین عملیات delete کردن یک گره است و شما باید گره خواسته شده را حذف کنید. دومین عملیات insert کردن یک گره است. سومین عملیات جست و جوی گره است و شما باید نتیجه جست و جو را به صورت True/False برگردانید. در نهایت هم لازم است که پیمایش preorder درخت را چاپ کنید.

### ورودی

در خط اول ورودی تعداد راس های BST مشخص شده. در خط دوم ورودی پیمایش postorder درخت به شما داده شده. در خط سوم یک گره به شما داده شده که باید آن را از درخت حذف کنید. تضمین می شود گره مشخص شده در درخت وجود دارد. در خط چهارم یک گره به شما داده شده و باید این گره را در درخت insert کنید. در خط آخر یک گره به شما داده شده و باید این گره را در درخت جست و جو کنید.

19

17 18 22 35 30 36 48 43 26 20 5 79 62 59 83 53 92 94 86

30

89

30

## خروجی

در خط اول باید نتیجه جست و جو گره داده شده در خط پنج ورودی را به صورت True یا False برگردانید.

در خط بعد نیز پیمایش preorder درخت را پس از انجام این عملیات ها چاپ کنید.

False

86 53 5 20 18 17 26 22 43 36 35 48 83 59 62 79 94 92 89

## ▪ نکات مهم

- تضمین می شود درخت ها هیچگاه گره تکراری نخواهند داشت.
- برای حذف یک گره در صورتی که هم فرزند راست و هم فرزند چپ داشته باشد، مطابق اسلایدهایتان می توان predecessor یا successor آن را جایگزینش کرد. برای حل این سوال شما همواره predecessor گره را جایگزینش کنید.



## • آلیس در مغازه گل فروشی (۱۵ نمره)

حین گشت‌وگذار در سرزمین عجایب، آلیس سر از مغازه گل‌فروشی در می‌آورد. درست مثل هر چیز دیگری در سرزمین عجایب این مغازه نیز غیر عادیست. صاحب مغازه به آلیس نحوه خرید را توضیح می‌دهد:

"در این مغازه تعداد زیادی گلدان مشاهده می‌کنی که در هر کدام تعدادی گل وجود دارد. به ازای هر سکه که خرج می‌کنی می‌توانی تمام گل‌های یک گلدان را برداری. بعد از برداشتن تمام گل‌های یک گلدان، نصف (در صورت زوج بودن) یا کف نصف (در صورت فرد بودن) تعداد گل‌های چیده شده دوباره در گلدان رشد می‌کنند و تو می‌توانی با پرداخت سکه بعدی، دوباره از این گلدان هم استفاده کنی."

آلیس تعداد سکه‌های محدودی دارد و می‌خواهد همه سکه‌هایش را برای خرید گل خرج کند. به او کمک کنید تا بیشترین تعداد گل ممکن را از این گلدان‌های عجیب بردارد.

## ورودی

در گل‌فروشی  $N$  گلدان وجود دارد. در خط اول ورودی هم  $N$  عدد با space از هم جدا شده‌اند که هر عدد نشان دهنده تعداد گل‌های موجود در یک گلدان است و می‌تواند بین  $1$  تا یک میلیارد باشد. در خط دوم ورودی تعداد سکه‌های آلیس (coin) مشخص شده است.

74 90 22 34 3 82 61 11

7

## خروجی

در تنها خط خروجی از شما انتظار می‌رود باقی مانده حداکثر تعداد گل‌هایی که آلیس می‌تواند بخرد را بر عدد  $1000000003$  چاپ کنید. (زیرا حداکثر تعداد گل‌هایی که آلیس می‌تواند بخرد ممکن است آنقدر بزرگ شود که از ظرفیت متغیرها بیشتر شود!)

430

### ▪ نکات مهم

- برای گرفتن نمره کامل ابتدا داده ساختار Heap را پیاده‌سازی کنید. سپس از Heap برای حل سوال استفاده کنید. استفاده از کتابخانه‌های آماده به جای پیاده‌سازی Heap مجاز نیست.
- به محدوده ورودی ها و زمان داده شده دقت کنید. حواستان باشد پیچیدگی زمانی راه حل‌تان زیاد نشود تا به ارور Time Limit برخورد نکنید.

❖ برای مشاهده مثال های بیشتر و درک بهتر سوال ها به سامانه کوئرا مراجعه کنید.