

به نام خدا

تمرین سری سوم بخش اول

تاریخ ارسال 1399/8/25

تاریخ تحویل 1399/9/7

- 1- برای تمرین های عملی خود می توانید از هر زبانی که مسلط تر هستید استفاده نمایید.
- 2- تمرین های خود را حتما از طریق hws ارسال نمایید.
- 3- تمرین های خود را زیپ کرده و نام فایل زیپ خود را به فرمت زیر تغییر دهید.

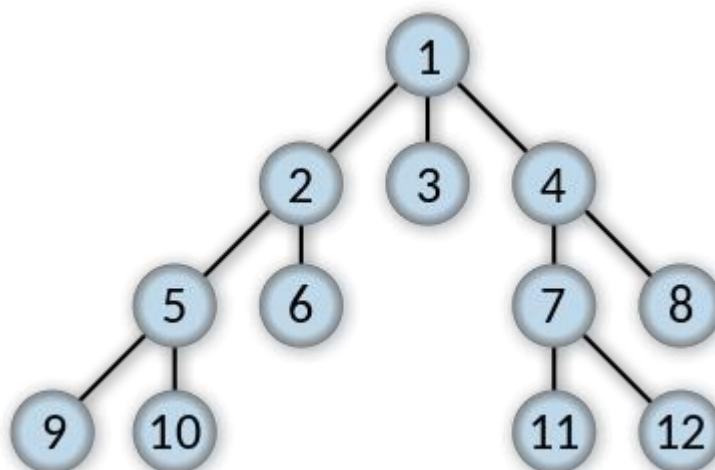
DS-991-firstname_lastname-name project

تمرین تئوری

مسئله ۱.

[5 نمره]

درخت عمومی زیر را به یک درخت دودویی تبدیل نمایید.



مسئله ۲.

[5 نمره]

نمایش پسوندی درخت عبارتی به صورت $// - 9 + 5678! * 4 - 123 \circ$ داده شده است. نمایش پیشوندی درخت را بیابید (ارقام، عملوندهای درخت و کلیه عملگرهای آن، به جز ! دوتایی هستند).

مسئله ۳.

[15 نمره]

فرض کنید دو درخت دودویی جستجوی متوازن T_1 و T_2 به ترتیب دارای n_1 و n_2 رأس، مجموعه‌های S_1 و S_2 را نمایش می‌دهند.

الف) الگوریتمی ارائه دهید که در مرتبه زمانی $O(n_1 \log(n_2))$ و با استفاده از حافظه اضافی $O(1)$ مشخص کند که آیا $S_1 \subseteq S_2$ است یا خیر.

ب) الگوریتمی ارائه دهید که در مرتبه زمانی $O(n_1 + n_2)$ و با استفاده از حافظه اضافی $O(n_1 + n_2)$ مشخص کند که آیا $S_1 \subseteq S_2$ است یا خیر.

ج) الگوریتمی ارائه دهید که در مرتبه زمانی $O(n_1 + n_2)$ و با استفاده از حافظه اضافی $O(\log(n_1) + \log(n_2))$ مشخص کند که آیا $S_1 \subseteq S_2$ است یا خیر.

نکته : $S_1 \subseteq S_2$ یعنی مجموعه S_2 ، مجموعه S_1 را در بر داشته باشد.

مسئله ۴.

[15 نمره]

یک درخت جستجوی دودویی متوازن با n راس و ارتفاع $O(\log n)$ را در نظر بگیرید. با تغییر ساختار درخت و ذخیره یک سری اطلاعات اضافه در گره ها، داده ساختاری طراحی کنید که هر کدام از اعمال زیر را در زمان $O(\log n)$ انجام دهد:

- $insert(x)$: عضو x را به داده ساختار اضافه کند.
- $delete(x)$: عضو x را از داده ساختار حذف کند.
- $interval(x, y)$: تعداد عناصر درخت که در محدوده x تا y قرار دارد را برگرداند.

راهنمایی: در هر راس درخت، تعداد عناصر موجود در زیردرخت آن را نگه دارید...

مسئله‌ی ۵.

[10 نمره]

ثابت کنید در یک د.د.ج از هر گره‌ای شروع کنیم، می‌توانیم k عنصر بعدی آن را (در صورت وجود) با $O(h + k)$ بیابیم که h ارتفاع درخت است.

تمرین عملی

مسئله‌ی ۱.

[25 نمره]

متدی بنویسید که با دریافت نتیجه پیمایش Preorder یک درخت Postorder آن را برگرداند.
مثال:

120 100 90 80 35 32 30 40 : ورودی

خروجی : 35 32 30 120 100 90 80 40

مسئله ۲.

[25 نمره]

آرایه ای به طول N دریافت کرده و با آن یک درخت دودویی جستجو تشکیل دهید.
سپس دو عدد A و B که در آرایه وجود داشته باشد را دریافت کرده و بزرگترین عدد در مسیر A و B را به عنوان خروجی نمایش دهید.
مثال:

ورودی : $arr[] = \{ 18 \ 36 \ 9 \ 6 \ 12 \ 10 \ 1 \ 8 \}$

$A = 1$

$B = 10$

خروجی : 12

