

مستندات پروژه پایانی ساختمان داده

یاسمن روحانی فر - ۹۲۲۱۳۰۹۹

جزئیات طراحی ساختمان داده :

این ساختمان داده متشکل از ۳ درخت AVL است.

درخت AVL کلاسی است که از درخت Binary Search Tree (BST) ارث بری می کند.

علت استفاده از AVL به جای BST این است که برای insert و delete کردن یک Node پیچیدگی زمانی AVL خیلی بهتر از پیچیدگی زمانی BST است . (برای BST این پیچیدگی $O(n)$ و برای AVL $O(\log n)$ است !)

Node درخت ها خود کلاسی است که از یک Key و یک Value تشکیل شده است.

در این ۳ درخت AVL پیاده سازی شده تنها Key های Node ها هستند که با هم متفاوتند و همه ی Value ها در Node ها ثابت است.

Value کلاسی است که تمام مشخصات یک فرد را در بر می گیرد.

علت در نظر گرفتن ۳ ساختمان داده برای پیاده سازی راحت تر findbyid , findbyname , findbybirthrange است . چون Node ها در درخت ها بر اساس Key هایشان قرار می گیرند.

پیچیدگی زمانی عملیات خواسته شده (Worst Case) :

۱. افزودن یک فرد جدید به بانک اطلاعاتی هنگام تولد : $O(|L| \cdot \log n)$

که $|L|$ طول بزرگترین رشته ی موجود در درخت است.

۲. حذف یک فرد از بانک اطلاعاتی هنگام فوت $O(|L| \cdot \log n)$

۳. تغییر نام یک فرد : $O(|L| \cdot \log n)$

۴. یافتن مشخصات یک فرد :

۱-۴ : بر اساس شناسه (ID) : $O(|L| \cdot \log n)$

۲-۴ : بر اساس نام : $O(|L| \cdot \log n)$

۵. یافتن فامیل های نسبی درجه ۱ یک فرد : $O(|L| \cdot \log n) + O(C)$

که C تعداد فرزندان فرد است.

۶. یافتن فامیل های نسبی درجه ۲ یک فرد : $O(|L| \cdot \log n) + O(K+S+S^2)$

که K تعداد عمو ها و خاله ها , S تعداد نوه های فرد و S^2 تعداد cousin های فرد است.

۷. یافتن تمام افراد متولد شده در یک دوره زمانی معین : $O(ILI * S)$

که S تعداد جواب ها و ILI طول رشته ی ماکسیمم است.

۸. بررسی آماری افراد (درصد زنان و مردان در جامعه) : $O(1)$

۹. چاپ نام همه ی افراد به ترتیب حروف الفبا : $O(ILI * \text{Logn})$

مقدار حافظه مورد نیاز برای ذخیره داده ها

۳ درخت داریم که در هر کدام n Node ذخیره شده است. اما مقادیر Value در Node های این ۳ درخت یکسان است ! در واقع Node ئه تمام این درخت ها به یک Info مربوط به خود اشاره می کنند و ما ۳ تا info برای ۳ درخت نداریم ! در حین اجرای عملیات و محاسبات مانند findrel1 و findrel2 و حافظه های دیگری در نظر گرفته می شوند که بعداً آزاد می شوند و برای ذخیره ی دائم داده ها به کار نمی روند . پس پیچیدگی مکانی این ساختمان داده n است که n تابع افراد است !