



دانشکده علوم ریاضی و آمار



نیمسال اول ۱۴۰۰-۱۴۰۱

مدرس: دکتر مجتبی رفیعی

ساختمان داده‌ها و الگوریتم‌ها

جلسه ۳۱

نگارنده: متین عسکری

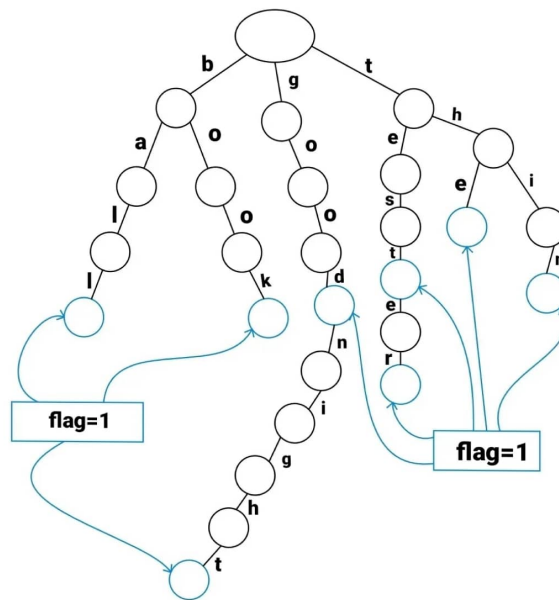
۷ آذر ۱۴۰۰

فهرست مطالب

۱	۱ ادامه مباحث درخت ترای (Trie)
۲	۲ عملیات روی درخت ترای
۲	۱.۲ عمل جستجو
۲	۲.۲ عمل درج
۳	۳.۲ عمل حذف

۱ ادامه مباحث درخت ترای (Trie)

مثال: برای سادگی تنها از اطلاع flag استفاده کرده و با اضافه کردن کلیدهای test و goodnight به مثال قبل، درخت ترای آن را بازترسیم می‌کنیم.



توجه: flag مابقی نودها صفر است (flag=0)

۲ عملیات روی درخت ترای

برخی از عملیات قابل تعریف بر روی درخت ترای در ادامه آورده شده است:

۱.۲ عمل جستجو

برای جستجوی یک کلید در درخت ترای نیاز است که حداکثر به تعداد کاراکترهای آن رشته، درخت را پیمایش کنیم بنابراین پیچیدگی جستجو از مرتبه $O(m)$ است. (جاییکه m تعداد کاراکترهای رشته مورد نظر است)

۲.۲ عمل درج

برای عمل درج یک کلید می‌بایست گام‌های زیر را دنبال کنیم:

- از ریشه شروع به پیمایش کرده و اگر کلید مورد نظر در پیمایش دیده‌شد، دو حالت پیش می‌آید:
 - * اگر flag آن ۱ بود، کلید از قبل در درخت ترای وجود دارد. (مثل tester در مثال قبل)
 - * اگر flag آن ۰ بود، flag را به ۱ تغییر می‌دهیم. (مثل test در مثال قبل)
- اگر کلید مورد نظر در پیمایش رؤیت نشد، می‌بایست نودهای بیشتری به طور مناسب بر درخت اضافه شود. (مثل اضافه کردن good night به مثال قبل)

نکته

پیچیدگی عمل درج از مرتبه $O(m)$ است، جاییکه m تعداد کاراکترهای رشته مورد نظر است.

۳.۲ عمل حذف

برای عمل حذف یک کلید می‌بایست گام‌های زیر را دنبال کنیم:

- از ریشه شروع به پیمایش کرده و اگر کلید مورد نظر رؤیت نشد، متوقف می‌شویم، مثل good by در مثال قبل.
- اگر کلید مورد نظر یافت شد و flag آن صفر بود، متوقف می‌شویم، مثل go.
- اگر کلید مورد نظر یافت شد و flag آن یک بود به صورت زیر عمل می‌کنیم:
 - اگر گره مربوطه دارای فرزند است می‌بایست flag گره فعلی را صفر کرد، مثل test در مثال قبل.
 - اگر گره مربوطه فاقد فرزند است می‌بایست علاوه بر حذف گره فعلی، نیاکان را به طور صحیحی نیز بروزرسانی کرد، مثل tester در مثال قبل.

سوال

دیکشنری را چگونه با یک درخت دودویی ذخیره کنیم به نحوی که امکان جست و جو در آن تا حد امکان کارا باشد؟

نکته تکمیلی: مقایسه کلی بین درخت k-تایی و درخت دودویی برای نگهداری دیکشنری:

- درخت k-تایی:
 - عمق کمتر ← سرعت بالا برای هرس کردن کلیدها
 - پهنای بیشتر ← فضای مصرفی بیشتر
- درخت دودویی:
 - عمق بیشتر ← سرعت هرس کردن کلیدها کمتر است
 - پهنای کمتر ← فضای مصرفی کمتر