

Trie adatszerkezet

Sergyán Szabolcs

2020. április 13.

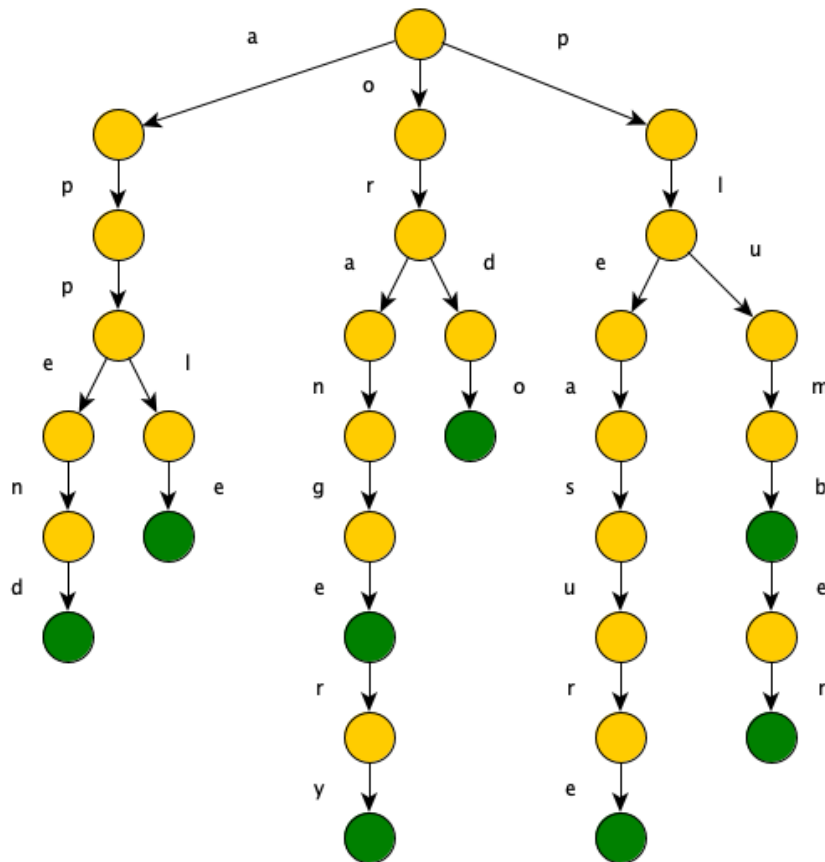
A trie¹ adatszerkezet alkalmas nagy mennyiségű szó eltárolására olyan módon, hogy egy új szó beszúrása és keresése is a szó hosszával arányos lépésben valósítható meg, azaz a futási idő nem függ a szótár méretétől.

A trie adatszerkezet egy gyökeres fa. Minden egyes csúcsában² összesen két dolgot tárolunk el:

1. az adott csúcs tekinthető-e szó végének, illetve
2. egy pointereket tartalmazó vektort (vagy tömböt), ami tartalmazza, hogy az egyes betűknek megfelelő csúcsok hol találhatók.

Az angol abc-t és csak kisbetűket használva a csúcsokban eltárolt vektor 26 pontert tartalmaz.

Az alábbi ábrán egy olyan trie látható, amely az *append*, *apple*, *orange*, *orangery*, *ordo*, *pleasure*, *plumb* és *plumber* szavakat tartalmazza.



¹A trie-nak nincs magyar megfelelője. A szó kiejtése kétféle módon is történhet az angol *tree* és *try* szavaknak megfelelően. Az adatszerkezet fa jellegű, melyben keresések (próbálkozások) végezhetők.

²angolul node

A zöld színű csúcsoknál véget érnek szavak, így ott a bool típusú jelölő igaz, a többi csúcsnál viszont hamis. Látható, hogy a zöld színű csúcsokból is lehet továbbhaladni, amennyiben van az adott szónál hosszabb szó is, aminek az eleje megegyezik egy szótárban lévő értelmes szóval.

A példából az is látható, hogy az egyes csúcsokban nem tárolunk el betűket. A betűket a csúcsokat összekötő élek reprezentálják. Persze ezek az élek valójában nem léteznek, hiszen az egyes csúcsokban eltárolt pointer vektorok felelnek meg nekik.

Amenyiben a szótárba egy új szót akarunk beszúrni, akkor a trie gyökeréből indulva tudjuk azt beszúrni. A szó egyes betűit a megfelelő csúcsban található pointer vektorokat használva szúrhatjuk be a fába. Például, ha a *banana* szót szeretnénk a például szolgáló trie-ba beszúrni, akkor először a gyökérben megvizsgálunk, hogy az ott tárolt pointer vektorban a *b* betűnek megfelelő elem mutat-e már valahova. Mivel még nem, hiszen nincs *b* betűs szó a trie-ban, ezért létrehozunk egy új csúcsot és a gyökér pointer vektorának *b*-t reprezentáló eleme ebbe az új csúcsba fog mutatni. A szó többi betűjével hasonlóan járunk majd el a megfelelő csúcsoknál. Amikor pedig a szó végére érünk, akkor a megfelelő csúcsnál a bool jelölőt igazra állítjuk.

Ha egy szótár minden eleméből felépítettünk egy trie-t, akkor nagyon gyorsan el tudjuk dönteni, hogy egy adott szó benne van-e a szótárban. A gyökérből kiindulva meg kell vizsgálnunk, hogy a szó egyes betűin végighaladva megtaláljuk-e a szót a szótárban.