

Matúš Lukáč – letný semester

Počas letného semestra som sa snažil implementovať aspoň prvú z dátových štruktúr. Konkrétne táto mala zakódovať pole čísiel (znakov) o veľkosti n nad abecedou o veľkosti t , pričom platí že $n \geq t$. Týmto som sa snažil dosiahnuť požadovane časy operácii a celkový priestor pre túto štruktúru, ktoré sú popísané v článku [Rank/select operations on large alphabets](#).

Pole som prepísal na tabuľku výskytov znakov: riadky reprezentujú jednotlivé čísla, stĺpce pozície. Následne sa tabuľka prepísala do row-major-order vektora, pomocou struktúry BitVector. Daný vektor som rozdelil na bloky o dĺžky t . Z neho vznikli dve štruktúry. Jeden vektor reprezentuje kardinalitu jednotlivých blokov: $1^{k_1}01^{k_2}...01^{k_n}$ kde k_i je počet jednotiek v danom bloku. Každý blok som potom zakódoval pomocou pozícií jednotiek v ňom – označme ich ako vektor E_i . Select vieme robiť potom v konštantom čase (iba prístup ku poli). Pre rank vyberieme každú $\lg(t)$ pozíciu z E_i , ktoré uložíme v Y-fast strome. Aby sme našli požadovanú hodnotu pre rank(j) v danom bloku spravíme binárne vyhľadávanie na E_i :

$$(\log t)rankFi(j) \leq rank-1_i(j) < (\log t)rankFi(j) + \log t$$

(rankFi()) je rank nad Y-fast stromom v danom bloku, rank-1_i() je rank v danom bloku)

Potom rank() a select() vieme implementovať nasledovne:

$$\begin{aligned} rank(j) &= rank-b(it) + rank-1_i(j-it) && (\text{kde } rank=b() \text{ je rank nad kardinálnym vektorom a } i = \lfloor j/t \rfloor) \\ select(j) &= select-1_i(j - rank-b((i-1)t)) + (i-1)t && (\text{kde } i = select-b(j), \text{ select-1}_i \text{ je select nad daným} \\ &&& \text{blokom, a select-b() je select nad kardinálnym vektorom.}) \end{aligned}$$

Môj projekt výrazne pokročil, aj keď sa mi ho nepodarilo celkom úspešne vyriešiť. Algoritmy a aj samotné dátové štruktúry vyžadujú ďalšie zdokonaľovanie, ladenie a debugging, aby sa problém úplne a presne vyriešil podľa daného článku.