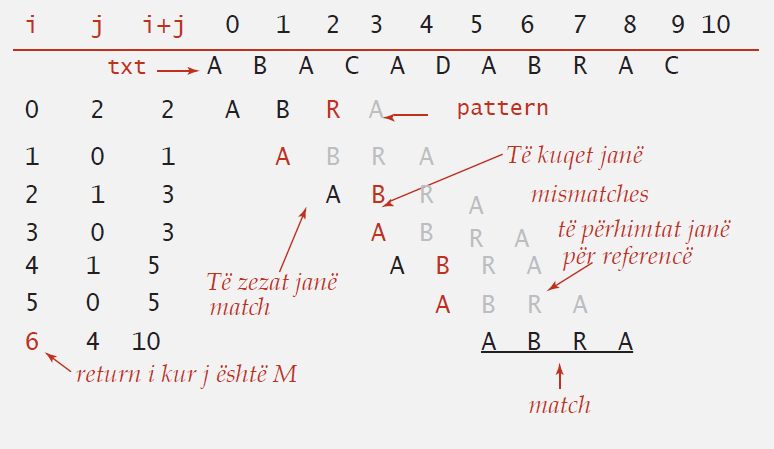
**SubString Search**

Patterni është pjes e Stringut e cila për nga gjatësia është shumë më i vogël se Stringu , patternin e shkruajm zakonisht me M ndërsa tekstin me N. Pra N >> M.

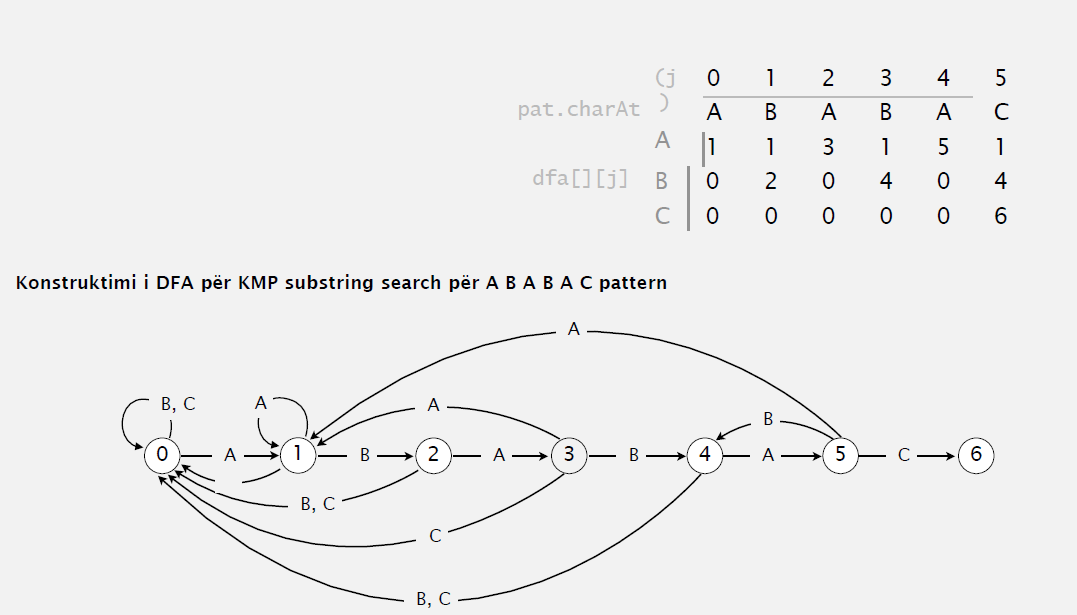
Aplikimet e SubStringSearch : Find & Replace, Forenzik kompjuterike për kërkim në memorie, identifikimi i spam emails, përgjime elektronike , screen scraping etj.

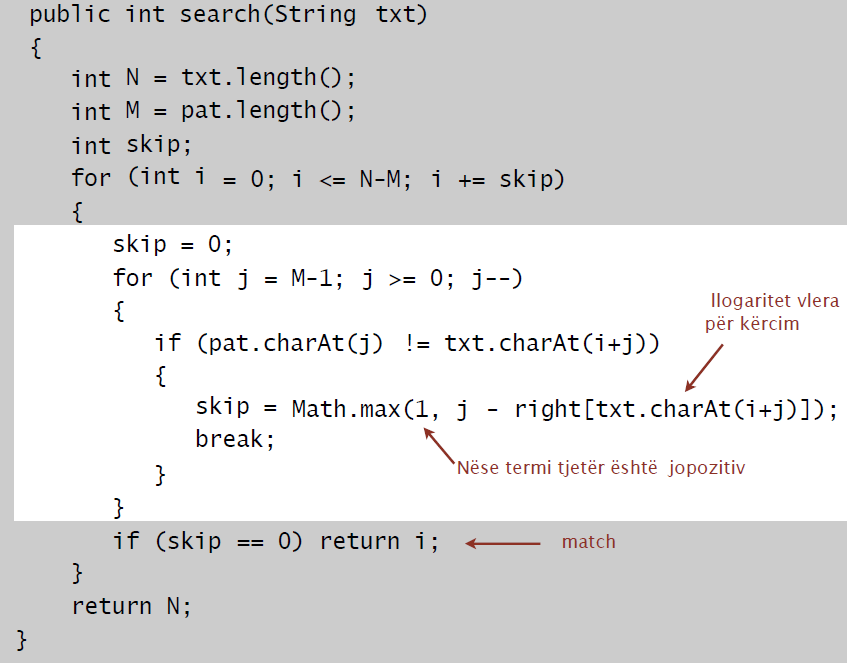
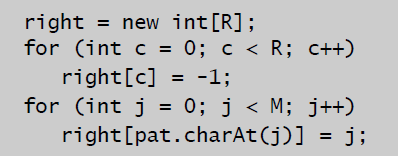
Ekzistojnë disa algoritme për substring search ndër të që do i përmendim janë : **BruteForce, Knuth-Morris-Pat, BoyerMoore, Rabin-karp.**

**BruteForce** : Algoritëm për substring search , punon me dy for loop-a , e gjen gjithnjë zgjidhjen e saktë , por me vonesë të madhe e kjo vones mund të shkoj në N\*M në rastin më të keq, rasti më i keq ndodhë kur në pattern dhe në tekst ka shumë karaktere që përsëriten. Edhe radixi i madh e ngadalëson shumë brute force.

Mangësitë e BruteForce : Nuk na garantonë kohë lineare dhe mban backup stream në tekst i cili e hargjon shumë memorien sidomos kur kërkojmë në stringje të gjata.

**Knuth-Morris-Pat :** Algoritëm për substring search , më i mirë se bruteforce , nuk ka nevojë për backup dhe implementohet me anë të DFA. Nuk e ka problem nëse ka karaktere të përsëritshme në tekst.

**DFA :** Maqinë absktrakte për suubstring search me një numër të fundme gjendiesh(states) , duke përfshi gjendjen fillestare(start) dhe atë të fundit (Halt) , për gjdo karakter që e gjen bën tranzision midis gjendieve, nëse arrin në gjendien e fundit (halt) kthen return , nëse jo nuk kthen asgjë nuk mund të bëjm accept. 

**Boyer-Moore :** Algortiëm për substring search , skenon tekstin nga e djathta në të majtë , punon me dy for loop , kodi ngjason me atë të brute force , por for loopa e par nuk bën nga një kalim por nga **skip** kalime, vlera e skip jepet me anë të formulës Math.max(1, j - right[txt.charAt(j+i)] , **right** është array në të cilën ruaj të vlerë -1 nëse karakteri nuk gjindet në pattern ose numër e pozitës së fundit në pattern për secilin karakter i cili gjindet në pattern.

**Rabin-Karp** : Algoritëm për substring search, ndryshe njihet si finger print algorithm , krahasimet i bën duke përdorur operacione aritmetike. Ky algoritem fillon duke e moduluar patternin me një numër, numër i cili duhet të jet dy shifra më i vogel se patterni dhe të jet numri më i madh i thjeshtë që ekziston me aq shifra. Pasi të kemi moduluar patternin fitojmë një vlerë e cila ruhet për tu krahasuar më vonë. Vazhdojmë për të moduluar tekstin në të cilin kërkojmë patternin , fillojmë duke përdorur metodën e hornerit , kjo metodë na thot të modulojmë së pari karakterin e parë e të vazhdojmë me të parin dhe të dytin , e kshtu me radhë deri të arrijmë madhësinë e patternit, pasi të arrijmë në kët pik do përodrim rolling hash , sepse rolling hash është e shpejt e saktë dhe e sigurtë , pra vazhdojmë duke moduluar M(gjatsia e paternit) karakteret e para , të dyta e kshtu me radhë deri të fitojmë një vlerë e cila është e barabartë me vlerën e fituar në rastin e modulimit të patternit.

Gjat modulimit numri nuk duhet të jet as shumë i vogël as shumë i madhë , në mënyr që tju shamgemi kollizoneve dhe overflow.

Rabin-Karp algoritmi i ka dy versione: **1.Monte Carlo Versioni :** Na kthen match nëse vlerat e hashit(e moduluara) janë të njëjta. **2.Las Vegas Versioni :** Nëse has në vlera të njëjta të hashit vazhdon me krahasime karakter mbas karakteri , nëse kemi substring match pra karakteret janë të njëjta na kthen match , nëse jo vazdhon kërkimin.

**Monte Carlo Versioni** : Na garanton gjithmonë kohë lineare por nuk na garanton gjithmonë përgjigje të saktë.

**Las Vegas Versioni** : Na garanton gjithmonë përgjigje të saktë po nuk na garanton gjithmonë kohë lineare , në rastin më të keq do kemi M\*N.

**Rabin-Karp Përparsitë :** Na mundëson të kërkojmë 2D patterns , dhe mundëson të kërkojmë patterna të shumfishtë në të njëjtën kohë.

**Rabin-Karp Mangësitë:** Operacionet aritmetike janë më të ngadalta sesa char krahasimet, Las vegas versionit i nevojitet back-up, në rastin më të keq granatojnë shumë pak.

