

ALGORITMOEN DISEINUA

Az8: Denetarik

- 1) $n \geq 1$ zenbaki naturala triangeluarra dela diogu baldin eta 1ean hasi eta jarraikiko zenbakiz osaturiko ordena gorakorreko sekuentzia ez-huts baten baturaren berdina bada. Ondorioz, lehenengo bost zenbaki triangeluarrak $1, 3=1+2, 6=1+2+3, 10=1+2+3+4$ eta $15=1+2+3+4+5$ dira. Idatz ezazu programa bat non emaniko zenbaki osoko positiboa zenbaki triangeluarra den ala ez erabakitzen duena. Programaren eraginkortasuna $O(n)$ -ren barnekoa izan behar du eta erabilitako memoria espazio estra konstantea. Zure soluzioaren ordena aztertu.
- 2) Izan bedi $V[1..n]$ erregistroen bektorea. Erregistroek kolorea izeneko eremu bat dute, non gorria, berdea edo urdina balioak har ditzaketen. Denbora lineala beharko duen algoritmo bat egizu V bektoreko balioak berrantolatzeke, kolore berdineko erregistroak jarraian gera daitezten. Zehazki, hasierako posizioetan gorri kolorekoak jarriko dira, ondoren berde kolorekoak eta azkeneko posizioetan urdinak.
- 3) Ordenaturik dagoen $V(1..n)$ bektorea emanik eta Z eta S bi balio (non $Z \leq S$), Z eta S balioen arteko balioz osaturiko V -ko azpibektorearen bi indizeak mugatzeko algoritmo bat egizu bilaketa dikotomikoaren ideia erabiliz. Soluzioak kasu txarrean $O(\lg n)$ ordena duela frogatu. Adibidez, sarrera $V=(3, 3, 3, 5, 5, 5, 7, 8, 8, 10, 12, 12, 12, 12, 12)$; $Z=5$ eta $S=8$ denean, irteerak $(4,9)$ izan behar du.
- 4) $G=(ERP, AR)$ grafo zuzendu pisudunak komunikazio sare bat adierazten du. $(u,v) \in AR$ arkuaren pisua $f(u,v)$ da, bere balioa $0 \leq f(u,v) \leq 1$ tartean egonik, u -tik v -raino komunikazio kanalaren fidagarritasuna adierazten du, hots, $u-v$ kanalaren ez hutsegitearen probabilitatea. Honela, $f(u,v)=0$ denean, kanala ez dela batere fidagarria adieraziko du, eta aldiz, $f(u,v)=1$ denean, komunikazio kanala erabat fidagarria dela adieraziko du. Demagun Xetik Yra doan kanalaren fidagarritasunaren huts egitearen probabilitatea kanala osatzen duten zatien probabilitateen biderkadura dela. Adibidez, $[x,y]$ bideak (x,z) eta (z,y) bi kanalak zeharkatzen dituen bidea bada, $[x,y]=(x,z)(z,y)$, orduan $f[x,y]=f(x,z)*f(z,y)$.
 A eta B erpinen arteko bide fidagarriena kalkulatzeko algoritmo eraginkor bat egizu. Soluzio eraikitzeke Dijkstraren soluzioan inspira zaitez.
- 5) Europako hainbat lurraldeetan “koipezko sukarra” epidemia pairatzen ari da. Gaixotasuna kutsatuta dauden abereekin kontatu zuzena edo zeharkakoa izanagatik

hedatzen da eta gainera, guztiz kutsakor, dela deskubritu dute. Infekzioa kontrolatzeko asmoz, granja guztietatik informazio bildu dute osasun autoritateek, $G=(g_1, g_2, \dots, g_n)$ izendatu direnak. Gainera, granjen artean egin diren abere salmentak eta hauen datak erregistratu dira.

Orain arte birusa granjen IG izeneko azpimultzo kaxkarrean detektatu da. Autoritateak kutsatuta egon litezkeen granjak identifikatu nahi ditu, berrogei eguneko itxialdia ezartzeko haietan. Kontuan izan behar da granja bat kutsatuta egoteko arriskua duela, baldin eta

- kutsatuta zegoen granja batetik abereak erosi baditu, edota
- kutsatuta egoteko arriskua zuen granja batetik abereak erosi baditu (hots, granja saltzailea kutsatua izan ostean saldutako abereak granja eroslea kutsa lezake).

Koarentenan aldarrikatu behar diren granjak zehaztuko dituen algoritmoa diseina eta azter ezazu.

- 6) Izan bedi A n osokoen taula, $n > 0$ izanik eta izan bedi *Batu?* funtzioa honako hau:

```
function Batu?(A, l, i ) return boolean is    -- i>0
    if i=1 and A(l) /= 0 then return false
    elseif A(i) = Batukaria(A, l, i-1) then return true
    else return Batu?(A, l, i-1)
```

non

```
function Batukaria(A, l, k ) return integer is
    b:=integer:=0
    for j:= 1 to k loop b:=b+A(j) end loop
    return b
```

Batu?(A, l, n) espresioaren ebaluazioak *true* itzultzen du A taulako osagai bat aurreko osagaien baturaren berdina den kasuan; eta *false*, baldintza hori betetzen duen osagairik existitzen ez duenean A taulan. Algoritmoaren eraginkortasuna aztertzea eskatzen zaizu. Erabil ezazu beste teknika bat problema bera eraginkorrago ebazteko. Soluzio berriaren eraginkortasuna aztertu.

- 7) Garraio enpresa batek M kaxez osatutako bidalketa garrantzitsu bat onartu du. Horretarako, N klaseetako ibilgailuak ditu. I klaseko ibilgailuek K_i kaxentzako kapazitate maximoa dute. I klaseko ibilgailu baten garraioak P_i prezio konstantea du, (ibilgailua beterik egon ala ez, eta hurbil joan ala ez). M kaxak kostu minimoan garraiatzeko **ibilgailuen antolaketa optimoa** kalkula ezazu programazio dinamikoan erabiliz. Proposatutako soluzioaren denbora-ordena eta erabilitako memoria espazio estra kalkula ezazu. Oharra: Garraioaren kostuaz gain, ez ahaztu ibilgailu mota bakoitzeko zenbat ibilgailu esleituko diren jakin nahi dela. Proposatutako soluzioaren denbora-ordena eta erabilitako memoria espazio estra kalkula ezazu.

- 8) Goranzko ordena jarraituz ordenatutako gizakien izenak dituzten $G_1(1..N)$ eta $G_2(1..N)$ bi bektore ditugu. Aldi berean bi bektoreetan agertzen diren gizakien izenak kalkulatu dituen programa idatzi behar da. Gizakien bektoreen luzera n denean, soluzioak har dezakeen denbora-ordena $T_t(n) \in \mathcal{O}(n \lg n)$ baina hobea izatea eskatzen da. Oharra: jatorrizko bektoretan izen errepikatuak baleude ere, ebakiduran ez lukete egon beharko.
- 9) Ausazko zenbakiak hamaika aplikagunetan dute: simulazioak, jokoak, segurtasun digitala, ... Konputagailu bidez ekoizten diren ausazko zenbakiak ez dira ausazko puruak. Ausazko aizunak (pseudo-ausazko) deritze eta formularen batetik ekoizten badira ere, estatistikoki aurretik ezin dira jakin. Ausazko zenbakiak modu sinplean ondorengo formula bidez konputa litezke. a , b eta m balioak emanik

$$x_{i+1} = (a * x_i + b) \text{ mod } m$$

hurrengo ausazko zenbakia sortzen du.

- a) Programazio dinamikoaren teknika bidez bai iteratiboki bai errekurtsiboki (memoriadun funtzioak erabiliz), lehenengo k ausazko zenbakiak itzuliko dituen algoritmoak egin itzazu, a , b , m eta x_0 hazia datuetatik.

Ausazko $((25, 34, 128), 51, 7) = [51, 29, 119, 65, 123, 37, 63]$
 non $a=25$, $b=34$, $m=8$ eta $x_0=51$ $K=7$

- b) Aurreko ataleko soluzioen konplexutasuna aztertu bai eta erabilitako memoria espazio estrea ere.

- 10) Demagun liburutegiaren arduraduna zarela. Jendeak egunero liburuak erabili ondoren bere lekuetan kokatu behar ditu, baina batzuetan nahasten dira eta toki okerrean uzten dituzte. Liburutegi osoaren tamainarekin alderaturik, gaizki kokaturik suertatzen diren liburuaren kopurua oso txikia da, eta gainera haien lekutik dezente gertu gelditzen ohi dira. Eguna bukatzean, liburu guztiak bere tokietan egotea lortu behar duzu.

- a) Klasean ikusitako algoritmoetatik zein uste duzu egokiena denik problema honi aplikatzeko? Zergatia azaldu, zure aukera beste algoritmoak baino hobea dela egiaztatuz (gutxienez beste hiru algoritmorekin konparatu behar duzu)
- b) Benetako liburutegietan hartutako liburuak birkokatzea erabiltzaileei ez zaie onartzen, eta arduradunek beraiek lan hori egitea nahiago izaten dute. Horren zergatia esplikatu (liburutegia ordenatuta mantentzearen kostuan oinarrituz).

- 11) X osagaia $S(1..N, 1..M)$ taula batean dagoen erabakiko duen programa egizu, non taulako lerroak eta zutabeak bektore ordenatuak diren. Soluzioaren $T(n, m)$ exekuzio denbora eta ordena bai eta erabilitako $MEE(n, m)$ mugatu.

- 12) G grafo ez-zuzendua emanik Kruskalen algoritmoak G -ren hedapen zuhaitz minimoa kalkulatu du Bilatu-Bateratu (*Union-Find*) datu-egitura erabiliz. Metodoak lehendabizi grafoaren ertzez osatutako multzoa haien pisu gorakorrean ordenatzen du, ondoren

ertz bakoitza banan bana aztertzeke. Kruskalen ondorengo bariante berria aztertzea eskatzen da. Aldaera berriak meta (HEAP) datu-egitura erabiliko du ertzak metatzeko, eta bertatik banan bana ertzak hartuko ditu ondoren iturri bertsioan bezala tratatzeko. Algoritmoa idaztea eta haren ordena kasu txarrenean kalkulatzeko eskatzen da. Erantzun biak justifikatzea derrigorrezkoa da.

13) Zirkuiting enpresak zirkuitu digitalak diseinatzen ditu eta euren propietateak aztertzeke programak bilakatzen hasi dira. Oraingoan, ondorengo osagaiak dituzten zirkuitu sinpleak aztertzeke programa bilakatzea eskatu du: pila baten polo positiboari eta polo negatiboari konektatutako kable pare, kableak konexio puntuak lotuz, bai eta kableak bonbillaren bi kontaktu puntuetara lotuak. Programak emaniko zirkuituak ondorengo hiru propietateko zein propietate betetzen duen itzuli behar du:

- a) Bonbilla piztuko da
- b) kortozirkuitua gertatuko da, pilaren polo positibotik negatibora eta bonbillatik pasa gabe korronea iritsiko delako.
- c) Bonbilla ez da piztuko korronea hura zeharkatzen ez duelako, baina kortozirkuitua ere ez da gertatuko korronea polo negatiboraino iristen ez delako.

14) Honako definizio errekursiboa ematen digute:

$$f(n, m) = \begin{cases} 0 & m = 0 \\ n & m = 1 \\ f\left(n, \left\lfloor \frac{m}{2} \right\rfloor\right) + f\left(n, \left\lceil \frac{m}{2} \right\rceil\right) & m > 1 \end{cases}$$

- Begi bistakoa da programazio errekursibo inozoaz (puruz) kodetuko balitz, kalkuluak errepikatuko lirakekeela. Hauek ekiditeko, programazio dinamikoaren teknika erabiliz implementa ezazu bai iteratiboki bai memoriadun funtzioak erabiliz, hots, bi moduetan.
- Aurreko ataleko soluzioen konplexutasun analisia egizu. Batuketa kopuruan diferentziarik dagoenik nabarmentzen al duzu?

15) n pertsonen artean, pertsona bat ospetsua da baldin eta denek hura ezagutzen badute baina hark inor ezagutzen ez badu. Pertsona ospetsurik existitzen den identifikatzeko ondoren galdera soilik uzten digute egiten: “Barkatu, hango pertsona hura ezagutzen duzu?” Galdera n pertsonetako edozeini egin badieziaiokegu eta jasotako erantzunak beti zuzenak direla suposatzen badugu, zein da egin egin ditzakegun galdera kopuru txikiena n pertsonen artean, haien arteko norbait ospetsua den edo ez mugatzeko?

16) n teorikok esamesak elkarraldatu nahi dituzte. Honela, telefono deiak 2 teoriko artean gertatzen badira, eta bi teorikok hitz egiten dutenean, biok dakizkiten esamesa guztiak elkarraldatzen badituzte, zenbat telefono dei egin beharko dira gutxienez n teorikoen esamesa guztiak jakin ditzaten?