OBLIKOVANJE I ANALIZA ALGORITAMA — 2. kolokvij

4. 7. 2008.

1. (20) Neka je $A_n = \{a_1, a_2, \ldots, a_n\}$ konačan skup različitih vrsta novčića. Vrijednosti novčića a_i su prirodni brojevi i vrijedi $a_1 > a_2 > \ldots > a_n$. Na raspolaganju imamo neograničenu količinu novčića svake vrste. Problem razmjene novca glasi: treba izabrati **najmanji** ukupni broj novčića tako da zbroj njihovih vrijednosti bude zadani prirodni broj C.

   (a) Ako je $a_n > 1$, pokažite primjerom da postoje A_n i C za koje problem nema rješenja.

   (b) Ako je $a_n = 1$, dokažite da rješenje postoji.

   (c) Za slučaj $a_n = 1$, sastavite pohlepni algoritam razmjene novca, koji prolazi novčiće redoslijedom $a_1, a_2, \ldots, a_n$, uzimajući maksimalni broj novčića određene vrijednosti a_i. Algoritam treba vratiti brojeve x_i novčića i–te vrste u razmjeni, za $i = 1, \ldots, n$. Nađite složenost ovog algoritma. Ovisi li ona o vrijednostima novčića a_i?

   (d) Nalazi li pohlepni algoritam **uvijek** rješenje s najmanjim brojem novčića? Posebno, što vrijedi za slučaj $a_i = k^{n-i}$, $i = 1, \ldots, n$, gdje je $k > 1$ zadani prirodni broj? Dokažite ili nađite kontraprimjer.

2. (20) Neka je p polinom stupnja $n = 2^k - 1$ s cjelobrojnim koeficijentima i vodećim koeficijentom (uz x^n) jednakim 1. Polinom p možemo napisati u obliku

$$p(x) = \left(x^{(n+1)/2} + a\right) q(x) + r(x) \quad ,$$

   gdje je a konstanta, a q i r su polinomi stupnja $2^{k-1} - 1$. Isti postupak rekurzivno primijenimo na polinome q i r. Na kraju, dobivamo polinom p izražen u terminima polinoma oblika $x^i + c_i$, gdje je i potencija od 2. Dobiveni oblik polinoma p zovemo **pripremljeni** oblik.

   (a) Izrazite polinom $p(x) = x^7 - 5x^6 + 4x^5 - 13x^4 + 3x^3 - 10x^2 + 5x - 17$ u pripremljenom obliku.

   (b) Neka je p zadan u pripremljenom obliku. Izaberite pogodnu strukturu podataka za ovaj prikaz polinoma. Sastavite algoritam koji računa vrijednost $p(x)$, za zadani $x \in \mathbf{Z}$. Nađite točan broj množenja i točan broj zbrajanja u tom algoritmu. Usporedite rezultate s brojem operacija u Hornerovom algoritmu. Kolika je ušteda?

OKRENITE!

3. (20) Magnetska traka sadrži n programa. Duljina programa i je l_i, za $i = 1, \ldots, n$. Vjerojatnost da s trake treba učitati program i je p_i, $i = 1, \ldots, n$. Pretpostavljamo da je

$$\sum_{i=1}^{n} p_i = 1 \quad .$$

Gustoća snimanja programa na traci je konstantna i brzina kretanja trake prilikom čitanja je konstantna. Svaki puta kada treba učitati neki program, traka se premotava na početak. Ako su programi na traci spremljeni u poretku i_1, i_2, $\ldots$, i_n, vrijeme potrebno za **čitanje** j-tog programa (s indeksom i_j) je

$$t_{i_j} = c \sum_{k=1}^{j} l_{i_k},$$

gdje je c neka konstanta (ovisi o gustoći pisanja na traku i brzini rada trake). **Prosječno** vrijeme potrebno za učitavanje programa je

$$T = \sum_{j=1}^{n} p_{i_j} t_{i_j} = c \sum_{j=1}^{n} p_{i_j} \sum_{k=1}^{j} l_{i_k}.$$

Redoslijed spremanja na traku je optimalan ako daje **minimalno** prosječno vrijeme T učitavanja programa.

(a) Pokažite primjerom da spremanje programa na traku u redoslijedu rastućem po l_i ne mora biti optimalno. Pokažite primjerom da spremanje programa na traku u redoslijedu padajućem po p_i, također, ne mora biti optimalno.

(b) Nađite redoslijed u kojem treba spremati programe na traku, tako da prosječno vrijeme učitavanja programa bude **minimalno**. Dokažite optimalnost tog redoslijeda.

(c) Sastavite algoritam koji nalazi optimalni redoslijed i nađite njegovu složenost.

4. (20) Ukratko opišite što radi diskretna Fourierova transformacija vektora duljine $n = 2^k$. Skicirajte rekurzivni algoritam za **brzu** diskretnu Fourierovu transformaciju (FFT) i izvedite njegovu složenost.

REZULTATI: ponedjeljak, 7. 7. 2008. u 9 sati. Saša Singer