OBLIKOVANJE I ANALIZA ALGORITAMA — 2. kolokvij

4. 2. 2009.

1. (20) U trgovinu, istovremeno, ulazi n kupaca. U toj trgovini kupce poslužuje samo jedan prodavač. On radi bez prestanka, a u danom trenutku može poslužiti točno jednog kupca. Kupci su numerirani brojevima od 1 do n. Za posluživanje i–tog kupca, prodavaču je potrebno vrijeme t_i. Vremena $t_1, t_2, \ldots, t_n$ su **unaprijed** poznata (zadana). Svaki kupac napušta trgovinu čim je poslužen. Treba naći optimalni redoslijed posluživanja svih n kupaca. To je onaj redoslijed koji minimizira ukupni boravak svih kupaca u trgovini, uključujući i vremena čekanja, ili, ekvivalentno, minimizira **prosječno** vrijeme boravka svih kupaca u trgovini.

   (a) Nađite optimalni redoslijed — odgovarajuću permutaciju brojeva od 1 do n, i dokažite optimalnost tog redoslijeda.

   (b) Sastavite algoritam koji, za zadani n i zadani niz $t_1, t_2, \ldots, t_n$, vraća optimalni redoslijed posluživanja i minimalno prosječno vrijeme boravka kupaca u trgovini.

   (c) Nađite složenost tog algoritma.

2. (20) Neka su a i b prirodni brojevi. Najveću zajedničku mjeru brojeva a i b nalazimo rekurzivnim Euklidovim algoritmom:

```
int gcd(int a, int b)
{
  if (b == 0) then
    return a;
  else
    return gcd( b, a % b );
}
```

   Mjera složenosti $T(a, b)$ ovog algoritma je broj izvršavanja operacije `%` (modulo), odnosno, broj (rekurzivnih) poziva funkcije `gcd`, bez polaznog vanjskog poziva.

   (a) Neka je $M = \max\{a, b\}$. Dokažite da za složenost u najgorem slučaju vrijedi

   $$T(a, b) \leq \lfloor 2 \lg M \rfloor + 1.$$

   Uputa: Ako je $a < b$ na početku, prvi korak algoritma samo "okreće" a i b, i to daje drugi član 1 (jednu dodatnu operaciju) na desnoj strani u gornjoj formuli. Zato uzmite da je na početku $a \geq b$. Pokažite da je tada

   $$a \bmod b \leq \frac{a-1}{2} \leq \frac{a}{2}.$$

   Zatim pogledajte što se događa nakon svaka 2 koraka algoritma.

   (b) Neka je F_n n–ti Fibonaccijev broj. Nađite složenost Euklidovog algoritma za računanje $\texttt{gcd}(F_n, F_{n-1})$. Koliko je $\texttt{gcd}(F_n, F_{n-1})$?

OKRENITE!

3. (20) Imamo n raznih predmeta numeriranih brojevima od 1 do n. Predmet i ima volumen w_i i vrijednost p_i. Ruksak volumena M treba napuniti nekim podskupom tih predmeta tako da ukupna vrijednost F svih predmeta u ruksaku bude maksimalna, a ukupni volumen svih predmeta ne prelazi zadani volumen ruksaka. Traženi podskup prikazujemo poljem x od n binarnih znamenki, gdje $x_i = 1$ označava da predmet i treba staviti u ruksak. Zadana ograničenja su:

$$w_i > 0, \quad p_i > 0, \quad x_i \in \{0, 1\}, \quad i = 1, \ldots, n.$$

Dodatno, pretpostavimo da je poredak predmeta kad ih sortiramo uzlazno po volumenu **jednak** poretku kad ih sortiramo silazno po vrijednosti.

(a) Dokažite ili opovrgnite primjerom sljedeću tvrdnju: pohlepno punjenje ruksaka po poretku iz gornje pretpostavke korektno rješava problem 0–1 ruksaka, tj. daje najveću moguću ukupnu vrijednost F predmeta koji stanu u ruksak.

(b) Sastavite algoritam koji, što je moguće brže, nalazi optimalno polje x i najveću ukupnu vrijednost F svih predmeta u ruksaku. Nađite složenost tog algoritma. Je li uvijek moguće potpuno napuniti ruksak?

4. (20) Ukratko opišite kako izgleda struktura disjunktnih skupova, koje osnovne operacije su definirane na toj strukturi i kako mjerimo složenost tih operacija. Opišite efikasne implementacije osnovnih operacija i komentirajte njihovu složenost.

REZULTATI: petak, 6. 2. 2009. u 12 sati.

Saša Singer