OBLIKOVANJE I ANALIZA ALGORITAMA — popravni kolokvij

11. 9. 2008.

1. (10) Između ponuđenih odgovora

$$\Theta(1), \quad \Theta(\lg\lg n), \quad \Theta(\lg n), \quad \Theta(n), \quad \Theta(n\lg n), \quad \Theta(n^2), \quad \Theta(2^n), \quad \Theta(n!),$$

nađite točan red veličine za broj koliko puta se izvršava naredba `x = x + 1` u svakom od sljedećih dijelova programa

(a)
```
i = n;
while (i >= 1) {
  x = x + 1;
  i = i / 2;
}
```

(b)
```
i = 2;
while (i < n) {
  i = i * i;
  x = x + 1;
}
```

2. (15) Zadana je rekurzivna relacija

$$T(n) = 5\,T(n/4) + f(n), \quad f(n) = n,$$

uz početni uvjet $T(0) = d > 0$. Nađite uvjetno asimptotsko ponašanje relacijom Θ za rješenje $T(n)$, ako je n potencija od 4. Da li se dobiveno rješenje može proširiti tako da asimptotsko ponašanje vrijedi bezuvjetno, za svaki dovoljno veliki $n \in \mathbf{N}$, ako u rekurziji piše $\lfloor n/4 \rfloor$, umjesto $n/4$?

3. (35) Zadano je polje T od n elemenata koje možemo međusobno uspoređivati relacijom $\leq$. Treba naći najmanji i najveći element u polju T. Ako najmanjih ili najvećih elemenata ima nekoliko, dovoljno je naći bilo koji. Elementarna operacija je uspoređivanje jednog para elemenata, s rezultatom logičkog tipa (laž, istina) . Mjera složenosti je točan broj takvih uspoređivanja (u ovisnosti o n).

(a) Sastavite algoritam `minmax_1` kao kombinaciju standardnih algoritama za nalaženje najmanjeg, odnosno, najvećeg elementa i nađite njegovu složenost.

(b) Sastavite algoritam `minmax_2` za rješenje ovog problema koji se rekurzivno primjenjuje na polovinama polja. Uz pretpostavku da je n potencija broja 2, nađite točno složenost ovog algoritma. Kako treba postupiti u slučaju da n nije potencija od 2?

(c) Obratite pažnju na "dno" rekurzije algoritma `minmax_2`. Može li se pažljivom realizacijom tog dijela algoritma dobiti **bolja** složenost nego za `minmax_1`?

Oba algoritma trebaju imati što manju složenost!

OKRENITE!

4. (35) 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.

5. (25) 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: petak, 12. 9. 2008. u 12 sati. Saša Singer