OBLIKOVANJE I ANALIZA ALGORITAMA — popravni kolokvij

30. 1. 2012.

1. (10) Neka su $f, g : \mathbf{N} \to \mathbf{N}$ dvije funkcije. Napišite preciznu definiciju asimptotske relacije ponašanja

$$f(n) \in o(g(n)) \quad (n \to \infty).$$

Koja svojstva ima relacija o (u smislu relacije uređaja)?

2. (10) Između ponuđenih odgovora

$$\Theta(1), \quad \Theta(\lg n), \quad \Theta(n), \quad \Theta(n \lg n), \quad \Theta(n^2), \quad \Theta(n^2 \lg n), \quad \Theta(n^3), \quad \Theta(2^n),$$

nađite točan red veličine za broj koliko puta se izvršava naredba `x = x + 1` u svakom od sljedećih dijelova programa (`/` je operator cjelobrojnog dijeljenja, kao u C-u):

(a)
```
i = n;
while (i >= 1) {
  for j = 1 to n
    x = x + 1;
  i = i / 3;
}
```

(b)
```
i = 1;
while (i <= n) {
  for j = 1 to i
    for k = 1 to n
      x = x + 1;
  i = i * 4;
}
```

Ukratko **argumentirajte** odgovore!

3. (35) Investicijska tvrtka planira buduća ulaganja na osnovu analize ponašanja dionica na tržištu u prethodnom razdoblju. U postupku analize često treba riješiti zadaće sljedećeg oblika. Promatra se ponašanje cijene određene dionice u bloku od n uzastopnih dana u prošlosti. Za svaki pojedini dan $i = 1, \ldots, n$, poznata je cijena c_i jedne dionice za taj dan (pretpostavimo da je cijena c_i konstantna kroz cijeli dan i).

Uzmimo da tvrtka u tom periodu želi **kupiti** 1000 dionica u nekom danu k, a zatim **prodati** sve te dionice u nekom **kasnijem** danu p. Na osnovu poznatih podataka treba naći koji dan je trebalo kupiti dionice i kada ih je trebalo prodati da zarada tvrtke bude **najveća** moguća. Ako u tom periodu od n dana **nije** bilo moguće ostvariti ikakvu zaradu, onda, umjesto traženih dana k i p, treba vratiti $k = p = 0$.

Na primjer, neka je $n = 3$, a poznate cijene jedne dionice kroz ta tri dana su $c_1 = 9$, $c_2 = 1$ i $c_3 = 5$ novčanih jedinica. Onda treba vratiti $k = 2$ (kupi na dan 2) i $p = 3$ (prodaj na dan 3), jer to daje zaradu od 4 novčane jedinice po svakoj dionici.

Sastavite algoritam koji, za zadani n i polje cijena c, nalazi optimalne dane k i p, ako oni postoje, odnosno, vraća $k = p = 0$, ako zarada nije moguća.

Red veličine složenosti algoritma mora biti $O(n \log n)$. Analizirajte složenost vašeg algoritma i pokažite da ona zadovoljava ovaj uvjet. Uputa: iskoristite princip "podijeli, pa vladaj" na polovinama zadanog vremenskog perioda (do $n/2$ dana i nakon toga).

OKRENITE!

4. (30) 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 $\texttt{gcd}(F_n, F_{n-1})$?

5. (35) Neka je $(A, \cdot)$ algebarska struktura polugrupe obzirom na binarnu operaciju množenja $\cdot : A \times A \to A$, tj. množenje je asocijativno i ima neutralni element $e \in A$. Za svaki element $a \in A$, korektno su definirane potencije a^n, za svaki $n \geq 0$. Složenost takvog potenciranja mjerimo brojem potrebnih množenja elemenata iz A.

(a) Napišite algoritam koji računa a^n, za zadane a i n, koristeći tzv. **binarno** potenciranje (pozicioni zapis broja n u bazi 2). Kolika je složenost tog algoritma, u ovisnosti o n?

(b) Pokažite primjerom da binarno potenciranje **ne mora** biti optimalno u pogledu broja množenja, tj. nađite neki n za koji se a^n može izračunati sa strogo manje množenja.

(c) Zadana je linearna homogena rekurzivna relacija reda k, s konstantnim koeficijentima

$$t_n = a_{k-1} t_{n-1} + \cdots + a_0 t_{n-k}, \quad n \geq k,$$

uz zadane početne uvjete $t_0, \ldots, t_{k-1}$. Ukratko opišite kako se binarno potenciranje može iskoristiti za **brzo** računanje n-tog člana t_n. Kolika je složenost u ovisnosti o n?

REZULTATI: srijeda, 1. 2. 2012. u 11 sati. Saša Singer