

Algoritamske strategije – test

17.10.2016.

Dat je algoritam za ispisavanje svih varijacija sa ponavljanjem brojeva od 1 do n dužine k . Varijacija sa ponavljanjem dužine k nad n elemenata je niz dužine k , pri čemu je svaki element niza jedan od n zadatih elemenata.

Npr. varijacije sa ponavljanjem dužine 2 nad brojevima od 1 do 3 su 1 1, 1 2, 1 3, 2 1, 2 2, 2 3, 3 1, 3 2, 3 3.

Rekurzivna funkcija za nalaženje svih varijacija:

```
void varijacije(int a[], int n, int k, int i)
{
    int j;
    if (i==k) ispis(a, k);
    else
        for (j=1; j<=n; j++)
        {
            a[i] = j;
            varijacije(a, n, k, i+1);
        }
}
```

Funkcija za ispis niza:

```
void ispis(int a[], int k)
{
    int i;
    for (i=0; i<k; i++)
        printf("%d ", a[i]);
    printf("\n");
}
```

U funkciji **varijacije**, niz a predstavlja jednu varijaciju, a brojač i trenutnu poziciju u varijaciji. Funkcija radi tako što postavi svaki mogući broj od 1 do n za prvi element varijacije. Za svaki mogući prvi element se rekurzivno poziva funkcija **varijacije** koja sada isprobava sve moguće brojeve na drugoj poziciji u varijaciji. Najzad, kada varijacija dostigne dužinu k , štampano je.

Za jedinicu mere vremenske kompleksnosti algoritma se uzima broj izvršavanja naredbi štampanja i broj poziva funkcije **varijacije**. Neka je f aritmetička funkcija kojom merimo vremensku složenost algoritma.

a) Neka poziv funkcije **varijacije** u glavnom delu programa izgleda ovako:

```
scanf("%d", &n);
k=2;
varijacije(a, n, k, 0);
```

Izračunati vrednost funkcije f u zavisnosti od parametra n . Dati ocenu gornje asimptotske granice (O) za funkciju f .

b) Neka poziv funkcije **varijacije** u glavnom delu programa izgleda ovako:

```
scanf("%d%d", &n, &k);
varijacije(a, n, k, 0);
```

Dati ocenu gornje asimptotske granice (O) za funkciju f koja sada zavisi od parametara n i k . Objasniti postupak.