

Ülesanne 1 (10 punkti). Milline on järgneva algoritmi $K(a, b, c, s_a, t_a, s_b, t_b, s_c, t_c, n)$ asümptootiline tööaeg (n suhtes; loeme, et n on K väljakutsel arvu kaks aste)?

```

if  $n = 1$  then  $\{c_{s_c, t_c} := c_{s_c, t_c} + a_{s_a, t_a} \cdot b_{s_b, t_b};$  return  $\}$ 
 $K(a, b, c, s_a, t_a, s_b, t_b, s_c, t_c, n/2);$ 
 $K(a, b, c, s_a + n/2, t_a, s_b, t_b + n/2, s_c, t_c, n/2)$ 
 $K(a, b, c, s_a, t_a + n/2, s_b, t_b, s_c, t_c + n/2, n/2);$ 
 $K(a, b, c, s_a + n/2, t_a + n/2, s_b, t_b + n/2, s_c, t_c + n/2, n/2)$ 
 $K(a, b, c, s_a, t_a, s_b + n/2, t_b, s_c + n/2, t_c, n/2);$ 
 $K(a, b, c, s_a + n/2, t_a, s_b + n/2, t_b + n/2, s_c + n/2, t_c, n/2)$ 
 $K(a, b, c, s_a, t_a + n/2, s_b + n/2, t_b, s_c + n/2, t_c + n/2, n/2);$ 
 $K(a, b, c, s_a + n/2, t_a + n/2, s_b + n/2, t_b + n/2, s_c + n/2, t_c + n/2, n/2)$ 

```

Ülesanne 2 (25 punkti). Olgu meil antud kirjed võtmeväärtustega k_1, \dots, k_n (järjestatud kasvavalt), me tahame need kirjed organiseerida kahendotsimise puuna. Meil on teada, et seda puud pärast loomist enam ei muudeta, samuti teame me, et kõigis otsimispäringutes, mis selles kahendotsimise puus tehakse, on otsitav võti üks võtmetest k_1, \dots, k_n . Me teame ka veel (iga i jaoks), et päringuid, kus otsitavaks võtmeks on k_i , on a_i tükki. Neist teadmistest lähtuvalt tahame me antud kirjeid sisaldava kahendotsimisipuu moodustada sellise, et sealt kirjete leidmine võimalikult vähe aega võtaks. Mingi kirje leidmise aeg sõltub tema asukohast puus, täpsemalt öeldes tema kaugusest puu juurtipust — puu juurtipus asuva kirje leidmine võtab aega ühe ajaühiku, juurtipu vahetus alluvas asuva kirje leidmine kaks ajaühikut, jne. Kui l_i on i -ndat kirjet sisaldava tipu kaugus kahendotsimise puu juurest, siis me soovime minimeerida suurust $\sum_{i=1}^n a_i l_i$.

Kirjuta programm, mis lähtudes sagedustest a_1, \dots, a_n eelkirjeldatud omadusega kahendotsimise puu konstrueerib. Vastuseks olevat puud võib kujutada näiteks nii, et lugeda, et meil on antud kirjemassiiv r_1, \dots, r_n , millel on väljad *vasak* ja *parem*. Programm peab siis need väljad õigetele kirjetele viitama panema ja puu juure tagastama.

Vihje. Vaata alamülesandeid kujul „Antud kirjed võtmeväärtustega k_i, k_{i+1}, \dots, k_j ja sagedustega a_i, a_{i+1}, \dots, a_j . Leia optimaalne kahendotsimise puu“. Siin $1 \leq i \leq j \leq n$.

Ülesanne 3 (20 punkti). Tõesta formaalselt järgmise programmi korrektsus (eel- ja järeltingimus on antud vastavalt enne ja pärast programmi; $sort_n(a)$ tähendab, et elemendid $a[1], \dots, a[n]$ on sorteeritud mittekahanevalt).

```

 $\{n \geq 1 \wedge sort_n(a)\}$ 
 $l := 1$ 
 $r := n$ 
 $t := false$ 
while  $l < r \wedge \neg t$  do
   $k := \lfloor \frac{l+r}{2} \rfloor$ 
  if  $a[k] = z$  then
     $t := true$ 
  else
    if  $a[k] < z$  then
       $r := k - 1$ 
    else
       $l := k + 1$ 
if  $l = r$  then
   $t := (a[l] = z)$ 
else
  skip
if  $l > r$  then
   $t := false$ 
else
  skip
 $\{t = true \Leftrightarrow \exists i \in \{1, \dots, n\} : a[i] = z\}$ 

```