

OHJ-2010 Tietorakenteiden käyttö

Tentti 17.5.2010

Terhi Kilamo

Tentissä ei saa käyttää ylimääräistä kirjallista materiaalia, laskimia, tietokoneita tai muita lunttausvälineitä.

Muista vastata kaikkiin tehtäviin.

Kirjoita vastauksesi siistillä käsialalla lyhyesti - vastauksia ei arvostella viivoittimella.

Vääristä vastauksista ei yleisesti vähennetä pisteitä, mutta tentin tarkastaja pidättää itsellään mahdollisuuden antaa miinuspisteitä täysin järjettömistä tai sisäisesti ristiriitaisista vastauksista (siis selvistä arvauksista).

1. Selitä lyhyesti (max.3 riviä/kohta) seuraavat käsitteet.

- iteraattorin mitätöityminen (1 p)
- assosiatiivinen säiliö (*associative container*) (1 p)
- hajasaanti-iteraattori (*random access iterator*) (1 p)
- Järjestä alla oleva taulukko käyttäen annettua algoritmia. Kirjoita, mitä algoritmi tulostaa. Kirjoita myös alaindeksit osoittamaan samanarvoisten alkioiden keskinäistä järjestystä. (3 p)

10	5 ₁	2 ₁	5 ₂	2 ₂	9	1	3
----	----------------	----------------	----------------	----------------	---	---	---

```

MERGE-SORT(A, p, r)
  if p < r then
    q := [(p + r)/2]
    MERGE-SORT(A, p, q)
    MERGE-SORT(A, q + 1, r)
    MERGE(A, p, q, r)
  PRINT(A)

```

```

MERGE(A, p, q, r)
  for i := p to r do
    B[i] := A[i]
  i := p
  j := p; k := q + 1
  while j ≤ q and k ≤ r do
    if B[j] ≤ B[k] then
      A[i] := B[j]
      j := j + 1
    else
      A[i] := B[k]
      k := k + 1
    i := i + 1
  if j > q then
    k := 0
  else
    k := q - r
  for j := i to r do
    A[j] := B[j + k]

```

2. Pitävätkö seuraavat väittämät paikkansa? (0.5 p/kohta)
- a) Jos algoritmin suoritusaika on kertaluokassa $\Theta(n)$, se on varmasti myös kertaluokassa $\Omega(n)$. ✓
 - b) Jos algoritmin suoritusaika on kertaluokassa $\Theta(n)$, se on varmasti myös kertaluokassa $O(n)$. ✓
 - c) Jos algoritmin suoritusaika on kertaluokassa $\Omega(\lg n)$, se on varmasti myös kertaluokassa $\Theta(\lg n)$. ✓
 - d) Jos algoritmin suoritusaika on kertaluokassa $O(\lg n)$, se on varmasti myös kertaluokassa $\Theta(\lg n)$. ✓
 - e) Jos algoritmin suoritusaika on kertaluokassa $O(n)$, se on varmasti myös kertaluokassa $O(n \lg n)$. ✓
 - f) Jos algoritmin suoritusaika on kertaluokassa $\Omega(n)$, se on varmasti myös kertaluokassa $\Omega(n \lg n)$. ✓
 - g) Jos algoritmin suoritusaika on kertaluokassa $\Theta(n)$, se on varmasti myös kertaluokassa $O(\lg n)$. ✓
 - h) Jos algoritmin suoritusaika on kertaluokassa $\Theta(n)$, se on varmasti myös kertaluokassa $\Omega(\lg n)$. ✓
 - i) Hajautustaulun koon tulisi olla samaa luokkaa kuin avainten arvoalueen. ✓
 - j) Alkion poistaminen hajautustaulusta on kertaluokassa $O(n)$. ✓
 - k) Alkion etsiminen keosta avaimen perusteella on kertaluokassa $O(\log n)$. ✗
 - l) Suurimman alkion etsiminen keosta on kertaluokassa $\Theta(\log n)$. ✗
3. a) Kuvaile kolme erilaista puurakennetta. Anna jokaisesta esimerkki, millaisessa tilanteessa kyseistä puuta voisi käyttää. (3 p) ^{EB}
- b) Alkion lisääminen STL:n vectorin loppuun on *tasatusti* (amortisoidusti) vakioaikaista, mutta keskelle lisääminen on lineaarinen operatio. Perustele, miksi. Selitä myös, mitä tasatusti vakioaikainen operatio tarkoittaa. (3 p)

4. a) Mitä alla oleva algoritmi tekee ja mikä on sen suoritusaika O - ja Ω -merkinnöillä ilmaistuna, kun säiliö S on

- i. pino?
- ii. jono?
- iii. keon päälle rakennettu prioriteettijono?

(3 p)

```
FOO(  $n$ ,  $target$  )  
  for  $i := 1$  to  $n$  do  
    read  $tmp$   
     $S.push(tmp)$   
  for  $j := 1$  to  $n$  do  
     $S.pop(x)$   
    print  $x$   
    if  $x = target$  then  
      return
```

- b) Olet menossa treffeille ja tarvitset asuusi mustat sukat. Olet eilen pessyt koneellisen mustia sukkia, jotka ovat nyt myttynä kuivausrummussa. Kuinka kauan sinulla menee O ja Ω -merkinnöillä ilmaistuna löytää pari sukkamytystä. Huomaa, kaikille mytyn sukille ei välttämättä edes ole paria - ainakaan mytyssä.
(3 p)

5. Edessäsi on labyrintti ja sinun pitää päästä labyrintin läpi mahdollisimman nopeasti. Kuvaile millaista tietorakennetta ja tietotyyppäjä käyttäisit labyrintin esittämiseen C++-ohjelmassa. Millaista algoritmia käyttäisit reitin hakuun? Miten tilanne muuttuisi, jos osalla labyrintin käytävistä olisi örkkejä, joita vastaan taisteleminen hidastaisi matkaa. Selosta, miten hakualgoritmisi etenevät labyrintissa. (6 p)

