

08.71.23/24 — Tölvunarfræði 2/2a

Vorpróf 11. maí 2007.

Engin hjálpargögn eru leyfileg.

Öll dæmi gilda jafnt.

Munið að skrifa notkunarlýsingu með forskilyrði og eftirskilyrði fyrir sérhvert stef og fastayrðingu gagna fyrir sérhverja útfærslu gagnamóts.

Athugið vel: Svara þarf tilskildum fjölda dæma úr hverjum hluta prófsins. Að því skilyrði uppfylltu gilda **10 bestu dæmi** til einkunnar. Þið hafið því 18 mínútur fyrir hvert dæmi að meðaltali. Byrjið því á að svara dæmum sem krefjast stuttra svara og þið getið auðveldlega svarað.

Hluti I: Röksemdafærsla

Svarið a.m.k. 1 dæmi úr þessum hluta

1. Íhugið eftirfarandi lykkjumynstur:

```
// F
while( C1 ) {
    // I
    S1
    if( C2 ) break;
    S2
}
// E
```

Hvaða samband þarf að gilda milli F, C1, I, S1, C2, S2 og E til að lykkja þessi sé rökrétt?

2. Hvaða lágmarkssamband þarf að gilda milli lýsinga boða í undirklasa og yfirklasa til að tryggt sé að forrit sem nota þessa klasa séu rökrétt (miðað við að röksemdafærsla sé að öðru leyti rétt)?
3. Íhugið eftirfarandi lykkju, sem hefur þann tilgang að reikna margfeldi tveggja heiltalna:

```
// x,y >= 0
int p=0, q=x, r=y;
while( r != 0 ) {
    // x*y == p+q*r, r>=0
    ... (breytum hvorki x né y)
}
// ???
```

- Hvert er rökrétt eftirskilyrði lykkjunnar?
- Hvernig getum við forritað stofninn í lykkjuna á hraðvirkan hátt þannig að fjöldi umferða verði $O(\log y)$? (aðeins má deila eða margfalda með 2, engri annarri tölu).

Hluti II: Algrím

Svarið a.m.k. 3 dæmum úr þessum hluta

4. Fyllið inn þar sem spurningarmerkin eru, þ.e. skrifið hvað á að koma í stað ?1?, o.s.frv. Þetta eru samtals átta svör.

```
// Notkun: k = leita(f,i,j,x);
// Fyrir: f[i..j-1] er í minnkandi röð
// Eftir: f[i..j-1] er óbreytt, i <= k <= j, og
//          f[i..k-1] >= x > f[k..j-1]
int leita( double[] f, int i, int j, double x ) {
    if( i==j ) return i;
    int m = (i+j)/2;
    if( f[m] ?1? x )
        return leita(f,i,?2?,x);
    else
        return leita(f,?3?,j,x);
}

int leita( double[] f, int i, int j, double x ) {
    int p=i, q=j;
    while( ?4? ) {
        // | >=x | óþekkt | <x |
        // i      p          q      j
        int m = (?5?)/2;
        if( f[m] ?6? x )
            p = ?7?;
        else
            q = ?8?;
    }
    return p;
}
```

5. Gefið er eftirfarandi stef:

```
// Notkun: k = skipta(f,i,j);
// Fyrir: f[i..j-1] er ekki-tómt svæði í f
// Eftir: Búið er að víxla gildum í svæðinu þ.a.
// f[i..k-1] <= f[k] <= f[k+1..j-1]
int skipta( double[] f, int i, int j );
```

Skrifið quicksort stef (með lýsingu - notkun, fyrir og eftir) með hjálp þessa stefs. Ekki þarf að forrita skipta stefið.

6. Lýsið merge-sort og rökstyðjið að tímaflækja þess sé $O(n \log n)$.
7. Hverjar eru fastayrðingar lykkjunnar tveggja í heapsort? Rökstyðjið að tímaflækja heapsort sé $O(n \log n)$.
8. Lýsið fastayrðingum ytri lykkju í bæði selection-sort og insertion-sort. Rökstyðjið að tímaflækja beggja aðferða sé $O(n^2)$.

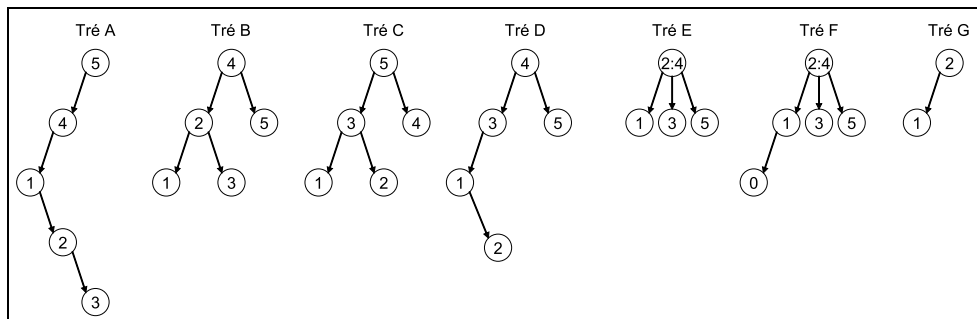
Hluti III: Gagnamót

Svarið a.m.k. 3 dæmum úr þessum hluta

9. Íhugið eftirfarandi myndir af trjám. Segið til fyrir hverja mynd hvort hún getur staðið fyrir eitt eða fleiri af eftirfarandi:
 - Tvíleitartré
 - AVL-tré
 - Splay-tré
 - 2-3 tré
 - Hrúga með hæsta gildi efst
 - Hrúga með minnsta gildi efst

Athugið að hér er ætlast til að öll tvíleitartré séu með gildin í vaxandi *in-order* röð.

Athugið einnig að sama mynd af tré getur vel staðið fyrir fleiri en eina af þessum upp töldu gerðum af trjám.



10. Hver er fastayrðing gagna fyrir AVL-tré?
11. Berið saman annars vegar splay-tré og hins vegar tætitöflur (hash tables) með tilliti til annars vegar hraða og hins vegar hvers konar aðgerðir er unnt að bjóða upp á. Hvor aðferðin er hraðvirkari? Hvor aðferðin gefur kost á fleiri mögulegum (hraðvirkum) aðgerðum? Gefið dæmi um aðgerðir sem unnt er að bjóða með annarri aðferðinni en ekki hinn.
12. Skrifðu klasa í Java eða C++ fyrir biðröð heiltalna. Þið megið sleppa því að forrita boðin, nema fyrir aðferðina til að bæta gildi í biðröðina, en munið að hafa notkun, forskilyrði og eftirsilyrði fyrir öll boð og munið að hafa skýra fastayrðingu gagna.
13. Skrifðu klasa í Java eða C++ fyrir hlaða heiltalna. Þið megið sleppa því að forrita boðin, nema fyrir aðferðina til að bæta gildi á hlaðann, en munið að hafa notkun, forskilyrði og eftirsilyrði fyrir öll boð og munið að hafa skýra fastayrðingu gagna.

Hluti IV: Blandað efni

Ekki þarf endilega að svara neinu dæmi úr þessum hluta, en ekki gleyma að svara 10 dæmum í heild.

14. Skrifðu forrit sem skrifar út 1000 minnstu prímtölurnar.
15. Skrifðu forrit sem les strengi frá aðalinntaki og skrifar strengina í vaxandi röð á aðalúttak.
16. Skrifðu stef sem finnur rót samfellds falls með helmingunarleit.
17. Skrifðu stef sem finnur lágsgildi samfellds falls með helmingunarleit.