

# 09.71.35 Stýrikerfi I

Upptökupróf

21. ágúst 1999  
kl. 9<sup>00</sup>-13<sup>00</sup>

Öll *skrifleg* hjálpargögn og reiknivél leyfileg.

- 1. [15%]** Harður diskur hefur 4 plötur og 8 leshausa. Á hverri hlið eru 13295 spor. Heildargagnamagn disksins er 17,28 GB (þar sem GB er í þessu tilfelli 1.000.000.000 bæti). Fjöldi geira er breytilegur, allt frá 230 til 384. Hver geiri inniheldur 512 bæti.
  - a) Hver er meðalfjöldi geira á spori, miðað við heildargagnamagn disksins?
  - b) Ef snúningshraði disksins er 5400 sn/mín, hver er þá hámarksflutningshraði frá disknum (miðað við að aðeins einn leshaus sé virkur í einu)?
  - c) Ef það tekur 0.9 msek að flytja leshausa milli aðliggjandi spora, hversu miklum flutningshraða er hægt að viðhalda í lengri tíma (e. sustained data rate)?
  
- 2. [10%]** Villur í geymslu og sendingu gagna koma oftast í runum, þannig að margir samliggjandi bitar skemmast, en síðan koma nokkuð margir bitar óskemmdir. Hvernig reynist Hamming villuleiðréttingaraðferðin við slíkar kringumstæður? Lýsið leið til að nota Hamming aðferðina þar sem allt að 100 samliggjandi bitar geta skemmt í einu.
  
- 3. [10%]** Eru til IEEE 32-bita fleytitölur  $a$  og  $b$  ( $b$  ekki núll), þannig að  $a + b = a$ ? Sýnið dæmi um slíkar tölur ef þær eru til, eða rökstyðjið að þær geti ekki verið til.
  
- 4. [20%]** Fallið hér að neðan ákvarðar hvort strengurinn  $S$  sé spegilstrengur (e. palindrome). Þið eigið að skrifa fallið, eins og það er hér að neðan (þ.e. endurkvæmt), sem 8086-smalamálsfall, sem kallanlegt er úr Borland C++.

```
int pal(char* S, int l, int r)
{
    if (l > r) return 1;
    if (S[l] == S[r])
        return pal(S, l+1, r-1);
    else
        return 0;
}
```

5. [10%] a) Lýsið nákvæmlega hvað eftirfarandi forritsbútur gerir. Gefið ykkur þær forsendur sem þið þurfið til að leysa verkefnið og takið þær fram.

```
vek      DW      33 DUP (OFFSET L1)
         DW      (256-33) DUP (OFFSET L2)
         . . .

L1:      xor      bx, bx
         mov      bl, [si]
         inc      si
         shl      bx, 1
         jmp      vek[bx]

L2:
```

b) Útskýrið nákvæmlega tilgang eftirfarandi forritsbúts. Inntakið er í gistingu AL og úttakið er það líka (Vísbending: Lítið á inntakið sem ASCII staf):

```
add      al, 40h
cbw
and      ah, 09h
add      al, ah
and      al, 0Fh
```

6. [15%] a) Við útreikninga á kvaðratrót í örgjörvum er gjarnan notuð ítrunaraðferð. Slíkar aðferðir byrja með nálgun á kvaðratrótinni og bæta hana síðan í hverri ítrun. Hvernig er hægt, á mjög hraðvirkan hátt, að finna nálgun á kvaðratrót fleytitölu á IEEE formi?

b) Skriðið forritsbút í 8086 smalamáli sem reiknar lausnir á jöfnunni  $ax^2 + bx + c = 0$ , þar sem  $a$ ,  $b$  og  $c$  eru fleytitölur í minni tölvunnar. Formúlan er

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Forritið ykkar þarf fyrst að athuga hvort  $b^2 - 4ac$  sé  $< 0$  og hoppa þá í einhvern villukóða (sem þarf ekki að skrifa). Ef svo er ekki þá eru lausnirnar tvær reiknaðar út og settar efst á fleytitölustaflann.

7. [10%] Skoðið eftirfarandi C++ forritsbút:

```
for (i=0, sum=0; i<1000000; i++) {
//    dummy = A[i + 16];
    sum += A[i + 0];
    sum += A[i + 1];
    sum += A[i + 2];
    sum += A[i + 3];
    sum += A[i + 4];
    sum += A[i + 5];
    sum += A[i + 6];
    sum += A[i + 7];
}
```

**(Dæmi 7 frh.)**

Þegar fyrsta línan í lykkjunni (þ.e. `dummy = ...`) er sett inn sem skipun (þ.e. athugasemdamerkið `//` tekið í burtu) þá eykst hraði forritsins um 25%. Gefið sannfærandi útskýringu á þessari hegðun og tiltakið þær forsendur sem þið gefið ykkur.

**8. [10%]** Þýðið eftirfarandi C-forritsbút yfir í PA-RISC smalamál. Þið megið nota breytunöfnin sem nöfn á gistum (t.d. `%x` fyrir `x`). Fallið `odd(i)` segir hvort inntakið sé oddatala eða ekki. Þið þurfið ekki að skrifa það sem fall.

```
x = 0;
y = 0;
for (i=0; i<n; i++)
    if ( odd(i) )
        for (j=i; j<n; j++) x += x+1;
    else
        for (j=0; j<i; j++) y += y+1;
```