Heimadæmi – heimadæmi 8

Arnar Sigurðsson

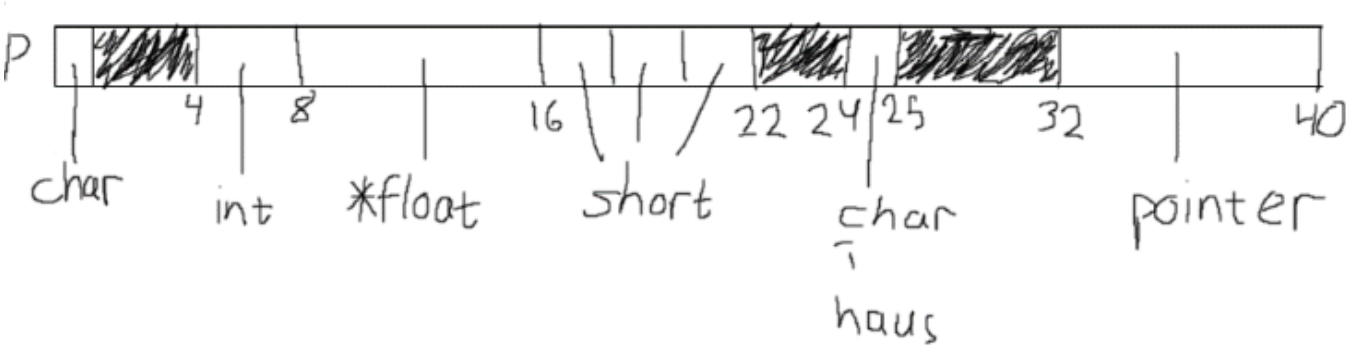
1. a) t = 5; og *t = 5;
   Í fyrri liðnum er gildið sett í gisti en í seinni lið er verið að setja gildið þangað sem gistið bendir á.

   b) Þarna er verið að forða gildinu sem er í %rbx með því að ýta því á hlaðann. Eftir skipunina er gildið áfram í %rbx en það er líka geymt þar sem %rsp var þegar skipunin var framkvæmd. Þegar %rsp fær gildið til sín er það sett á hlaðann og svo er %rsp lækkað um 8 (því hlaðinn vex niður á við).

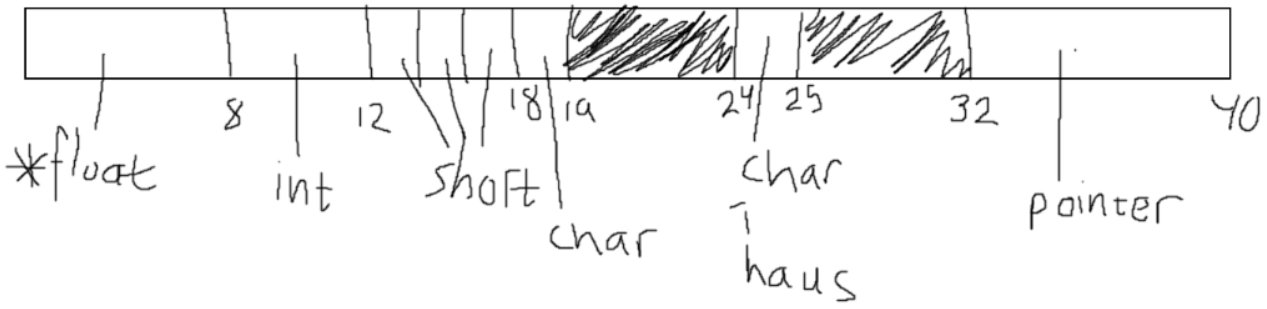
   c)

   d)

2. a) ónýtt svæði: 3 + 2 + 7 = 12



   b) Í rauninni ekki, þó raðað sé fyrri hlutanum hagkvæmt þá þarf rec_a haus að byrja í margfeldi af 8 og þar sem það þarf 19 hólf til að koma fyrri gögnum fyrir byrjar rec_a haus í sæti 24 og fyrsta sætið er char svo pointerinn sem þarf 8 bæti þarf þá að byrja í hólfi 32 og upp í 40. Þar með er það jafnlangt og jafnmörg ónýtt svæði eða 12.



3. a)

```
[ars98@krafla heimadaemi8]$ ./variation
0x7ffe3ee6fc68
[ars98@krafla heimadaemi8]$ ./variation
0x7fffee4fcda8
[ars98@krafla heimadaemi8]$ ./variation
0x7ffea3aff248
[ars98@krafla heimadaemi8]$ ./variation
0x7ffdbebed918
[ars98@krafla heimadaemi8]$ ./variation
0x7ffd88532fb8
[ars98@krafla heimadaemi8]$ ./variation
0x7fffe23d8928
```

Kemur greinilega á slembinn hátt.

b)Minnsta virðist vera 0x7ffe01745ba8 og hæsta virðist vera 0x7ffffd7295a8, munurinn er 0x1fbfe3a00 eða í decimal: 8522709504 svo þónokkur munur.

4. a) fun1:

```
3         mov     eax, 0
4         call    f
5         mov     rbx, rax
6         mov     eax, 0
7         call    f
8         mov     rdx, rax
9         lea     rax, [rbx+rbx*2]
10        lea     rbx, [rax+rdx*2]
11        mov     eax, 0
12        call    f
13        add     rax, rbx
```

fun2:

```
18        mov     eax, 0
19        call    f
20        lea     rax, [rax+rax*2]
21        add     rax, rax
```

b) Ef til dæmis fallið f() skilar víðværri eða global breytu þá uppfærist hún við kallið á fun1 og uppfærða breytan er svo notuð í fun2 sem gefur þá ekki sömu niðurstöðu.

5.

```
1  addto:
2          test    rdx, rdx
3          jle     .L1
4          mov     eax, 0
5  .L3:
6          mov     rcx, QWORD PTR [rsi+rax*8]
7          add     QWORD PTR [rdi+rax*8], rcx
8          add     rax, 1
9          cmp     rdx, rax
10         jne     .L3
11 .L1:
12         ret
```

Fyrst er athugað hvort rdx, inntak 3 = n sé 0 eða minna því þá er listinn tómur eða invalid og þá er hoppað yfir allt. Svo er sótt stakið í fylkinu w og það sett í rcx. Því næst er því bætt við fylkið v á viðeigandi stað í fylkinu sem er þar sem fylkið v byrjar (rdi) + fjöldi ítranna(rax) * 8 (hvert stak í fylkinu er 8 bæti). Svo er ítrað um einn, athugað hvort það er komið upp í lengdina á fylkjunm n, ef svo er er returnað en annars er farið annan hring í lykkjunni.