

## Laboration 2 – Intro till Simulatoren

1. Först reserverar vi två minnesutrymmen á 32-bitar styck (word). Dessa ger vi namnet var1 respektive var2. Därefter ser vi till att assemblern inte får ordna om våran kod.

Vi talar om att efter detta så kommer våra instruktioner till programmet.  
Vi sätter 'start' till en global etikett samt talar om att detta skall vara den etikett som körs vid start.

'start:' härefter ger vi programmets instruktioner vid start.  
addi \$8,\$0,17 adderar talet 17 med innehållet i \$0 (alltså 0) till register 8.  
addi \$9,\$0,9 adderar talet 9 med innehållet i \$0 (alltså 0) till register 9.

Härpå sparar vi nu det vi har i register 8 till minnet, detta genom var1 som är en etikett till ett word. Samma sak med register 9.

Därpå avslutas programmet.

2. På adresserna 80020000 till 80020014
4. På adresserna 00020000 till 00020014
5. Eftersom assemblern översätter vissa "fake"-operationer till riktig assemblerkod.
6. 00020020 till 00020024
7. Högerklicka och välj FLOAT i snabbmenyn.
8. mult1:  
    .word 0  
sum:  
    .word 0  
  
    nop  
    add \$11,\$8,\$9  
    mult \$8,\$9  
    mflo \$10  
  
sw \$10,mult1  
sw \$11,sum

9.  $y=x+5$ ; motsvarar

lw \$2,x  
addu \$3,\$2,5  
sw \$3,y

#laddar in innehållet i x till \$2  
#adderar innehållet i \$2 med 5 och lägger i \$3  
#sparar innehållet i \$3 till y

och

$z=y+x$ ; motsvarar

lw \$2,y  
lw \$3,x  
addu \$2,\$2,\$3  
sw \$2,z

#laddar in innehållet i y till \$2  
#laddar in innehållet i x till \$3  
#adderar \$2 med \$3 och sparar i \$2  
#sparar innehållet i \$2 till z