Laboration 3

Bilaga A

HENRIK BÄCK
850611-6253

Karlstads Universitet
2005-02-14

Handledare: Hans Hedbom
Nils Dåverhög

```
#include<iregdef.h>

.data

ANTAL:
           .int 10

v:
           .int 4
           .int 5
           .int 2
           .int 2
           .int 1
           .int 6
           .int 7
           .int 9
           .int 5
           .int 10

string1:
           .asciiz " %d"

nyrad:
           .asciiz "\n"

.set noreorder
.text
.globl start
.ent start

start:

           la a0,v
           jal skriv
           nop
           la a0,v
           move a1,zero
           lw a2,ANTAL
           nop
           addi a2,a2,-1
           nop
           jal QuickSort
           nop
           la a0,v
           nop
           jal skriv
           nop
           jal _exit
           nop

.end start

.globl skriv
.ent skriv

skriv:
           subu sp,sp,16
           sw ra,12(sp)
           sw s0,8(sp)
           sw a0,4(sp)
           la a0,nyrad
           nop
           jal printf
           nop

           move s0,zero

L1:
           lw t2,ANTAL
           nop
           bge s0,t2,klar
           nop
           la a0,string1
           sll t0,s0,2
           la t1,v
           nop
           add t1,t1,t0
           nop
           lw a1,0(t1)
           jal printf
           nop
           la a0,nyrad
           nop
           jal printf
           nop
           addi s0,1
           b L1
           nop

klar:
           lw a0,4(sp)
           lw s0,8(sp)
           lw ra,12(sp)
           addiu sp,sp,16
           nop
           jr ra
           nop

.end skriv

.globl Partition
.ent Partition
Partition:

# t0 = pivot = v[a]
# t1 = lower = a+1
# t2 = upper = b
# t3 = v[lower]
# t4 = v[upper]
# t5 = temp
# t6 = adresshållare
# t7 = adresshållare
# t8 = tempplats
# t9 = adress till till v[a]
# a0 = v[]
# a1 = a
# a2 = b

           sll t0,a1,2
           nop
           add t0,a0,t0
           nop
           move t9,t0
           lw t0,0(t0)
           addi t1,a1,1
           move t2,a2

P1:


           sll t3,t1,2
           nop
           add t3,a0,t3           #adress till v[lower]
           nop
           move t6,t3             #sparar undan adress till senare
           lw t3,0(t3)
           nop
           bgt t3,t0,P2           #if v[lower] > pivot goto P2
           nop
           blt t2,t1,P2           #if upper < lower goto P2
           nop
           addi t1,t1,1           # lower = lower+1
           b P1
           nop

P2:

           sll t4,t2,2
           nop
           add t4,a0,t4           #adress till v[upper]
           nop
           move t7,t4             #sparar undan adress till senare
           lw t4,0(t4)
           nop
           bge t0,t4,P3           #if pivot => v[upper] goto P3
           nop
           blt t2,t1,P3           #if upper < lower goto P3
           nop
           addi t2,t2,-1
           b P2
           nop

P3:

           blt t2,t1,P4           #if upper < lower goto P4
           nop
           move t5,t3             #temp = v[lower]
           lw t8,0(t7)            #t8 = v[upper]
           nop
           sw t8,0(t6)                       #v[lower] = v[upper]
           sw t5,0(t7)                       #v[upper] = temp
           addi t1,t1,1
           addi t2,t2,-1

P4:
           bge t2,t1,P1           #if upper >= lower goto P1
           nop
           sll t4,t2,2
           nop
           add t4,a0,t4           #adress till v[upper]
           sll t3,t1,2
           nop
           add t3,a0,t3          #adress till v[lower]
           move t7,t4            #spara undan adress
           move t6,t3            #spara undan adress
           lw t4,0(t4)                      #v[upper]
           lw t3,0(t3)                      #v[lower]
           nop
           move t5,t4            #temp = v[upper]
           sw t0,0(t7)                      #v[upper] = pivot
           sw t5,0(t9)                      #v[a] = temp
           move v0,t2            #return upper

           jr ra
           nop

.end Partition

.globl QuickSort
.ent QuickSort
QuickSort:

           subu sp,sp,32
           sw ra,28(sp)
           sw a0,24(sp)
           sw a1,20(sp)
           sw a2,16(sp)
           sw s0,12(sp)
           sw s1,8(sp)

           bge a1,a2,Q1          #if a>=b goto Q1
           nop
           jal Partition         #retreve k
           nop
           move s0,v0            #k = Partition(v,a,b)
           nop
           addi s0,s0,-1         #k = k -1
           move s1,a2            #sparar a2
           move a2,s0
           jal QuickSort
           nop
           move a2,s1            #lägg tillbaka värde
           addi s0,s0,2          #k = k+1
           move s1,a1            #sparar a1
           move a1,s0
           jal QuickSort
           nop
           move a1,s1            #lägg tillbaka värde

Q1:

           lw ra,28(sp)
           lw a0,24(sp)
           lw a1,20(sp)
           lw a2,16(sp)
           lw s0,12(sp)
           lw s1,8(sp)
           addi sp,sp,32

           jr ra
           nop

.end QuickSort
```