

# ARMAssemblyInt:SmallEx:A

Prg5.s

```
01 .text
02 .align 2
03 .global max
04 .syntax unified
05 max:
06 @ link register save eliminated.
07 str fp, [sp, #-4]!
08 mov fp, sp
09 sub sp, sp, #8
10 str r0, [fp, #-4]
11 str r1, [fp, #-8]
12 ldr r2, [fp, #-8]
13 ldr r3, [fp, #-4]
14 cmp r2, r3
15 movge r1, r2
16 movlt r1, r3
17 mov r0, r1
18
19 mov sp, fp
20 ldr fp, [sp], #4
21 bx lr
22 .size max, .-max
23 .align 2
24
25 .global maxOf3
26 .fpu vfp
27 .type maxOf3, %function
28 maxOf3:
29 stmfd sp!, {fp, lr} @ or push {fp, lr}
30 add fp, sp, #4
31 sub sp, sp, #12
32 str r0, [sp, #8]
33 str r1, [sp, #4]
34 str r2, [sp]
35 ldr r1, [fp, #-12]
36 ldr r0, [fp, #-8]
37 bl max
38 mov r3, r0
39 ldr r1, [fp, #-16]
40 mov r0, r3
41 bl max
42 mov r3, r0
43 mov r0, r3
44 sub sp, fp, #4
45 ldmdf sp!, {fp, pc} @ or pop {fp, pc}
46
47 .section .rodata
48 .align 2
49 .LC0:
50 .ascii "%d%d%d\000"
51 .align 2
52 .LC1:
53 .ascii "Max is %d\012\000"
54 .text
55 .align 2
56 .global main
57 .type main, %function
58 main:
59 push {fp, lr}
60 add fp, sp, #4
61 sub sp, sp, #16
62 sub r3, fp, #20
63 sub r2, fp, #16
64 sub r1, fp, #12
65 ldr r0, .L7
66 bl __isoc99_scanf
67 ldr r3, [fp, #-12]
68 ldr r1, [fp, #-16]
69 ldr r2, [fp, #-20]
70 mov r0, r3
71 bl maxOf3
72 str r0, [fp, #-8]
73 ldr r1, [fp, #-8]
```

str fp, [sp], #-4 @ Save fp to stack  
str lr, [sp], #-4 @ Save lr to stack

**Less Efficient and Longer:**

```
cmp r2, r3
bge r2IsBigger
mov r1, r3
b allDone
r2IsBigger:
mov r1, r2
allDone:
```

