

0.1 Sprintf

```
# A mini-version of sprintf:  sprintf
# Date:  Feb. 8,2012

# Main program to test sprintf
    .data
buffer: .space 200 # 200 bytes, initialized to zero
newline:
    .ascii "\n"
ex1: .ascii "\n\nExample 1: should print 73\n"
ex2: .ascii "\n\nExample 2: should print 691\n"

    .text
main:
    li $v0,4 # system call for print_string
    la $a0,ex1
    syscall

    la $a0,buffer
    li $a1,73
    jal sprintf

    li $v0,4
    la $a0,buffer
    syscall

    li $v0,4 # system call for print_string
    la $a0,ex2
    syscall

    la $a0,buffer
    li $a1,691
    jal sprintf

    li $v0,4
    la $a0,buffer
    syscall
```

```

# Finally, exit:
    li $v0,10 # exit
    syscall

# end of main program

# sprintf(char *mybuff, int numval)

# Note: we will use $s1 for the pointer to the current position
#       in mybuff

sprintf:
    addi $sp, $sp, -8
    sw $ra, 4($sp)
    sw $s1, 0($sp)

    move $s1, $a0 # make $s1 pointer in mybuff

move $a0, $a1
jal dodecimal

    lw $s1, 0($sp)
    lw $ra, 4($sp)
    addi $sp, $sp, 8
    jr $ra

dodecimal:
    addi $sp, $sp, -8 #
    sw $ra, 4($sp) #
    sw $s0, 0($sp) # save the old value of $s0

    li $t9, 10 # constant, used for div and rem
    rem $s0, $a0, $t9 # value % 10 - the remainder
    addi $s0, $s0, '0' # convert that to a digit

    div $a0, $a0, $t9 # value / 10

```

```
        beqz    $a0, onedig    # is it non-zero?
        jal    dodecimal      # if so, print it

onedig: sb $s0, 0($s1) # put char in mybuff
        addi   $s1, $s1, 1 # move pointer

        lw     $s0, 0($sp)
        lw     $ra, 4($sp)
        addi   $sp, $sp, 8
        jr     $ra
```