

## IC220 SlideSet #5 -- SPIM

---

- 4 basic parts of a program
  1. Initialization - of variables
  2. Get Input - from program itself, from user, from file
  3. Do something - computing part, processing
  4. Give output

Example:

```
int main () {
    sum = 0;
    for (j=1; j<4 ; j++ )
        sum = sum + j;
    cout << sum << endl;
}
```

1

## Complete SPIM Program

*save in file with .asm extension*

---

```
.data
    cr: .asciiz "\n"

.text
    .globl main
main:
    li $a0, 0
    li $t0, 1
    li $t1, 4

loop: beq $t0, $t1, end
    add $a0, $a0, $t0
    addi $t0, 1
    j loop

end: li $v0, 1 # syscall #1 -- print int in $a0
    syscall    # does the print

    li $v0, 4 # syscall #4 - print string (address in $a0)
    la $a0, cr
    syscall    # does the print

    li $v0, 10 # terminate program
    syscall
```

< blank line at end of program >

2

## SysCalls

---

Service	System call code	Arguments	Result
print_int	1	\$a0 = integer	
print_float	2	\$f12 = float	
print_double	3	\$f12 = double	
print_string	4	\$a0 = string	
read_int	5		integer (in \$v0)
read_float	6		float (in \$f0)
read_double	7		double (in \$f0)
read_string	8	\$a0 = buffer, \$a1 = length	
sbrk	9	\$a0 = amount	address (in \$v0)
exit	10		
print_char	11	\$a0 = char	
read_char	12		char (in \$a0)
open	13	\$a0 = filename (string), \$a1 = flags, \$a2 = mode	file descriptor (in \$a0)
read	14	\$a0 = file descriptor, \$a1 = buffer, \$a2 = length	num chars read (in \$a0)
write	15	\$a0 = file descriptor, \$a1 = buffer, \$a2 = length	num chars written (in \$a0)
close	16	\$a0 = file descriptor	
exit2	17	\$a0 = result	

3

## DEMO

---

- Loading program – example.asm
- Running
- Reloading (**not** Reinitialize. Open file again if error)
- Stepping
- Setting Value
- Breakpoint
- Settings – decimal vs. hex
- SPIM help webpage
  - <http://www.cs.usna.edu/~lmcdowel/courses/ic220/spim/>

4