

CMSC 216

Introduction to Computer Systems

Lecture 15

Assembly Language, cont., and Procedures

Jan Plane & Alan Sussman
{jplane,als}@cs.umd.edu

Administrivia

- Project 3 due Monday
- Project 4 spec posted Tuesday, public tests soon after
- Finish reading Bryant and O'Hallaron Section 4.1 (Y86 subset) and Chapter 3, for more info on IA-32 instruction set architecture
- Read Section 3.7 of Bryant and O'Hallaron

Chapters 3 and 4.1, Bryant and O'Hallaron

ASSEMBLY LANGUAGE

Final Factorial Program

```
0x000: |          rdint  %ecx
0x002: |          irmovl $1,%ebx
0x008: |          irmovl $1,%eax
0x00e: |          rrmovl %eax,%edi
0x010: |          subl  %ecx,%edi
0x012: |          jg     0x2a

0x017: | Loop:      multl  %eax,%ebx
0x019: |          irmovl $1,%edi
0x01f: |          addl  %edi,%eax
0x021: |          rrmovl %eax,%edi
0x023: |          subl  %ecx,%edi
0x025: |          jle   0x17

0x02a: | EndWhile: wrint  %ebx
0x02c: |          irmovl $10,%edi
0x032: |          wrch  %edi
0x034: |          halt
```

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	0
%ecx	0
%edx	0
%ebx	0
%esp	0
%ebp	0
%esi	0
%edi	0

PC	0x000
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	0
%ecx	0
%edx	0
%ebx	0
%esp	0
%ebp	0
%esi	0
%edi	0

PC	0x002
CC: ZF	0
CC: SF	0
CC: OF	0

Input is: 4

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	0
%ecx	4
%edx	0
%ebx	0
%esp	0
%ebp	0
%esi	0
%edi	0

PC	0x002
CC: ZF	0
CC: SF	0
CC: OF	0

Input is: 4

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	0
%ecx	4
%edx	0
%ebx	0
%esp	0
%ebp	0
%esi	0
%edi	0

PC	0x008
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	0
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	0
PC	0x008
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	0
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	0
PC	0x00e
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	1
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	0
PC	0x00e
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	1
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	0
PC	0x010
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	1
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x010
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	1
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x012
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	1
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	-3

PC	0x012
CC: ZF	0
CC: SF	1
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	1
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	-3

PC	0x017
CC: ZF	0
CC: SF	1
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt

```

Register name	Value
%eax	1
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	-3

jg:
~(SF ^ OF) & ~ZF = 0

PC	0x017
CC: ZF	0
CC: SF	1
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt

```

Register name	Value
%eax	1
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	-3

PC	0x019
CC: ZF	0
CC: SF	1
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt

```

Register name	Value
%eax	1
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	-3

PC	0x019
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt

```

Register name	Value
%eax	1
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	-3

PC	0x01f
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	1
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x01f
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	1
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x021
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	2
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x021
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	2
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x023
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	2
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	2

PC	0x023
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	2
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	2

PC	0x025
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	2
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	-2

PC	0x025
CC: ZF	0
CC: SF	1
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	2
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	-2

PC	0x02a
CC: ZF	0
CC: SF	1
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt

```

Register name	Value
%eax	2
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	-2

jle:
(SF ^ OF) | ZF = 1

PC	0x017
CC: ZF	0
CC: SF	1
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt

```

Register name	Value
%eax	2
%ecx	4
%edx	0
%ebx	1
%esp	0
%ebp	0
%esi	0
%edi	-2

PC	0x019
CC: ZF	0
CC: SF	1
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt

```

Register name	Value
%eax	2
%ecx	4
%edx	0
%ebx	2
%esp	0
%ebp	0
%esi	0
%edi	-2

PC	0x019
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt

```

Register name	Value
%eax	2
%ecx	4
%edx	0
%ebx	2
%esp	0
%ebp	0
%esi	0
%edi	-2

PC	0x01f
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	2
%ecx	4
%edx	0
%ebx	2
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x01f
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	2
%ecx	4
%edx	0
%ebx	2
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x021
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	3
%ecx	4
%edx	0
%ebx	2
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x021
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	3
%ecx	4
%edx	0
%ebx	2
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x023
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	3
%ecx	4
%edx	0
%ebx	2
%esp	0
%ebp	0
%esi	0
%edi	3

PC	0x023
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	3
%ecx	4
%edx	0
%ebx	2
%esp	0
%ebp	0
%esi	0
%edi	3

PC	0x025
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	3
%ecx	4
%edx	0
%ebx	2
%esp	0
%ebp	0
%esi	0
%edi	-1

PC	0x025
CC: ZF	0
CC: SF	1
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	3
%ecx	4
%edx	0
%ebx	2
%esp	0
%ebp	0
%esi	0
%edi	-1

PC	0x02a
CC: ZF	0
CC: SF	1
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt

```

Register name	Value
%eax	3
%ecx	4
%edx	0
%ebx	2
%esp	0
%ebp	0
%esi	0
%edi	-1

```

jle:
(SF ^ OF) | ZF = 1

```

PC	0x017
CC: ZF	0
CC: SF	1
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt

```

Register name	Value
%eax	3
%ecx	4
%edx	0
%ebx	2
%esp	0
%ebp	0
%esi	0
%edi	-1

PC	0x019
CC: ZF	0
CC: SF	1
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt

```

Register name	Value
%eax	3
%ecx	4
%edx	0
%ebx	6
%esp	0
%ebp	0
%esi	0
%edi	-1

PC	0x019
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt

```

Register name	Value
%eax	3
%ecx	4
%edx	0
%ebx	6
%esp	0
%ebp	0
%esi	0
%edi	-1

PC	0x01f
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	3
%ecx	4
%edx	0
%ebx	6
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x01f
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	3
%ecx	4
%edx	0
%ebx	6
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x021
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	4
%ecx	4
%edx	0
%ebx	6
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x021
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	4
%ecx	4
%edx	0
%ebx	6
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x023
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	4
%ecx	4
%edx	0
%ebx	6
%esp	0
%ebp	0
%esi	0
%edi	4

PC	0x023
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	4
%ecx	4
%edx	0
%ebx	6
%esp	0
%ebp	0
%esi	0
%edi	4

PC	0x025
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	4
%ecx	4
%edx	0
%ebx	6
%esp	0
%ebp	0
%esi	0
%edi	0

PC	0x025
CC: ZF	1
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	4
%ecx	4
%edx	0
%ebx	6
%esp	0
%ebp	0
%esi	0
%edi	0

PC	0x02a
CC: ZF	1
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt

```

Register name	Value
%eax	4
%ecx	4
%edx	0
%ebx	6
%esp	0
%ebp	0
%esi	0
%edi	0

PC	0x017
CC: ZF	1
CC: SF	0
CC: OF	0

jle:
(SF ^ OF) | ZF = 1

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt

```

Register name	Value
%eax	4
%ecx	4
%edx	0
%ebx	6
%esp	0
%ebp	0
%esi	0
%edi	0

PC	0x019
CC: ZF	0
CC: SF	1
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt

```

Register name	Value
%eax	4
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	0

PC	0x019
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt

```

Register name	Value
%eax	4
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	0

PC	0x01f
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	4
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x01f
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	4
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x021
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	5
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x021
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	5
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x023
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	5
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	5

PC	0x023
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	5
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	5

PC	0x025
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	5
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x025
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	5
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x02a
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	5
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	1

jle:
(SF ^ OF) | ZF = 0

PC	0x02a
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	5
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	1

PC	0x02c
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	5
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	1

Output is: 24

PC	0x02c
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	5
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	1

Output is: 24

PC	0x032
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

0x000:	rdint	%ecx		
0x002:	irmovl	\$1,%ebx		
0x008:	irmovl	\$1,%eax		
0x00e:	rrmovl	%eax,%edi		
0x010:	subl	%ecx,%edi		
0x012:	jg	0x2a		
0x017:	multl	%eax,%ebx		
0x019:	irmovl	\$1,%edi		
0x01f:	addl	%edi,%eax		
0x021:	rrmovl	%eax,%edi		
0x023:	subl	%ecx,%edi		
0x025:	jle	0x17		
0x02a:	wrint	%ebx		
0x02c:	irmovl	\$10,%edi		
0x032:	wrch	%edi		
0x034:	halt			

Register name	Value
%eax	5
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	10

PC	0x032
CC: ZF	0
CC: SF	0
CC: OF	0

Output is: 24

Execution

0x000:	rdint	%ecx		
0x002:	irmovl	\$1,%ebx		
0x008:	irmovl	\$1,%eax		
0x00e:	rrmovl	%eax,%edi		
0x010:	subl	%ecx,%edi		
0x012:	jg	0x2a		
0x017:	multl	%eax,%ebx		
0x019:	irmovl	\$1,%edi		
0x01f:	addl	%edi,%eax		
0x021:	rrmovl	%eax,%edi		
0x023:	subl	%ecx,%edi		
0x025:	jle	0x17		
0x02a:	wrint	%ebx		
0x02c:	irmovl	\$10,%edi		
0x032:	wrch	%edi		
0x034:	halt			

Register name	Value
%eax	5
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	10

PC	0x034
CC: ZF	0
CC: SF	0
CC: OF	0

Output is: 24

Execution

0x000:	rdint	%ecx		
0x002:	irmovl	\$1,%ebx		
0x008:	irmovl	\$1,%eax		
0x00e:	rrmovl	%eax,%edi		
0x010:	subl	%ecx,%edi		
0x012:	jg	0x2a		
0x017:	multl	%eax,%ebx		
0x019:	irmovl	\$1,%edi		
0x01f:	addl	%edi,%eax		
0x021:	rrmovl	%eax,%edi		
0x023:	subl	%ecx,%edi		
0x025:	jle	0x17		
0x02a:	wrint	%ebx		
0x02c:	irmovl	\$10,%edi		
0x032:	wrch	%edi		
0x034:	halt			

Register name	Value
%eax	5
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	10

PC	0x034
CC: ZF	0
CC: SF	0
CC: OF	0

Output is: 24

Execution

0x000:	rdint	%ecx		
0x002:	irmovl	\$1,%ebx		
0x008:	irmovl	\$1,%eax		
0x00e:	rrmovl	%eax,%edi		
0x010:	subl	%ecx,%edi		
0x012:	jg	0x2a		
0x017:	multl	%eax,%ebx		
0x019:	irmovl	\$1,%edi		
0x01f:	addl	%edi,%eax		
0x021:	rrmovl	%eax,%edi		
0x023:	subl	%ecx,%edi		
0x025:	jle	0x17		
0x02a:	wrint	%ebx		
0x02c:	irmovl	\$10,%edi		
0x032:	wrch	%edi		
0x034:	halt			

Register name	Value
%eax	5
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	10

PC	0x035
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

```

0x000: | rdint %ecx
0x002: | irmovl $1,%ebx
0x008: | irmovl $1,%eax
0x00e: | rrmovl %eax,%edi
0x010: | subl %ecx,%edi
0x012: | jg 0x2a
0x017: | multl %eax,%ebx
0x019: | irmovl $1,%edi
0x01f: | addl %edi,%eax
0x021: | rrmovl %eax,%edi
0x023: | subl %ecx,%edi
0x025: | jle 0x17
0x02a: | wrint %ebx
0x02c: | irmovl $10,%edi
0x032: | wrch %edi
0x034: | halt
    
```

Register name	Value
%eax	5
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	10

PC	0x035
CC: ZF	0
CC: SF	0
CC: OF	0

Done!

Execution

- These are the final states of the PC, CC, and all the registers.
- Does this agree with what **yis** tells us?
- Remember, numbers are printed in hex, so $10_{10} = a_{16}$, $24_{10} = 18_{16}$

Register name	Value
%eax	5
%ecx	4
%edx	0
%ebx	24
%esp	0
%ebp	0
%esi	0
%edi	10

PC	0x035
CC: ZF	0
CC: SF	0
CC: OF	0

Execution

- Yes:

```

$ echo 4 | yis factorial.yo
24
    
```

Stopped in 34 steps at PC = 0x35. Exception 'HLT', CC Z=0 S=0 O=0
Changes to registers:

```

%eax: 0x00000000 0x00000005
%ecx: 0x00000000 0x00000004
%ebx: 0x00000000 0x00000018
%edi: 0x00000000 0x0000000a
    
```

Changes to memory:

Some other notes

- Exceptions
 - If your code exits due to a **halt** instruction, the exception will be HLT (which is fine)
 - You can also stop your code prematurely by passing a second parameter (max steps) to **yis**; this will usually result in the exception AOK
 - But if you manage to move to a bad address (forget to halt?), divide by zero, or just make the simulator otherwise unhappy, you'll get a different exception (e.g., ADR or INS)

Program efficiency

- Our assembly program was not as efficient as it could be
 - we could have optimized away the need for the variable **i** by simply decrementing **n** until **n** was 0
 - we also could have been smarter with our registers, and not overwritten the constant value we used repeatedly (1)
- These things need to be taken into consideration by compilers, and by you in the assembly code you're writing for the project...

Section 3.7, Bryant and O'Hallaron

PROCEDURE IMPLEMENTATION

Reminder: Y86 stack instructions

Instruction	Effect	Description
pushl R	$\text{Reg}[\%esp] \leftarrow \text{Reg}[\%esp] - 4;$ $\text{Mem}[\text{Reg}[\%esp]] \leftarrow \text{Reg}[R]$	Push on to stack
popl R	$\text{Reg}[R] \leftarrow \text{Mem}[\text{Reg}[\%esp]];$ $\text{Reg}[\%esp] \leftarrow \text{Reg}[\%esp] + 4$	Pop off of stack

- **pushl** and **popl** provide quick ways to work with the program stack
- Without **pushl**, a push operation would look like this:
irmovl \$4, %esi
subl %esi, %esp
rmmovl %eax, (%esp)

Y86 procedure-related instructions

Instruction	Effect	Description
call Label	push PC on stack $\text{PC} \leftarrow \text{Label}$	Call function
ret	$\text{PC} \leftarrow \text{pop value from stack}$	Return from function

- These two instructions are used to handle function calls in Y86
- **call** instruction pushes the return address on the stack, then jumps to the destination address
- **ret** instruction returns from a call
- functions are responsible for ensuring the stack is properly adjusted before returning

Quick example

- What is the value of `%eax` after this sequence of instructions?
`call next`
`next: popl %eax`
- Whatever the address of `next` (the `popl` instruction) is!
- Note: this is NOT the PC, but it is related to it
 - PC is the address of the next instruction to be executed; `%eax` now has address of most recently executed instruction (the `popl`)

Implementing functions

- When calling a function, the computer needs to know where execution will resume once the function returns (the *return address*)
- The `call` instruction stores this address on the stack; when the called function (the *callee*) is done, we'll need to undo any changes made to the stack so that we can get back to the caller

Call stack

- `%esp` contains the address of the top of the stack (the address of the element most recently pushed)
- The stack grows downward, from a maximum address of `0x1000`
- To use it, an assembly program must first initialize the stack pointer:
`irmovl $0x1000, %esp`
- Then, it can push and pop as necessary

Simple function call example

```
int n = 2;

void f();
void g();

int main() {
    f();
    n = 1;
    g();
    return 0;
}
void f() {
    g();
}
void g() {
    while (n > 0)
        printf("%d", n--);
}
```

```
main:    irmovl $0x1000, %esp
        call    f
        irmovl $1, %ebx
        rmmovl %ebx, n
        call    g
        halt
f:       call    g
        ret
g:       irmovl n, %ecx
Loop:   irmovl $0, %eax
        addl    %eax, %ecx
        jle     EndLoop
        writ    %ecx
        irmovl $1, %edx
        subl    %edx, %ecx
        rmmovl %ecx, n
        jmp     Loop
EndLoop: ret
n:       .align 4
        .long 2
```