

SPEC CODES EF 21	UWNET LOGIN NAME	Last, First Name (as in email and on scantron)
------------------------	------------------	--

Computer Sciences 354  
 Midterm Exam 2 Secondary  
 Thursday, Nov 7th, 2024  
 72 points (20% of final grade)  
 Instructors: Debra Deppeler, Hina Mahmood

1. **PRINT your UWNET Login Name** in box above.
2. **PRINT Last, First Name** in box above.
3. Use **#2 pencil** for all SCANTRON marks.
  - (a) **Record LAST NAME** Must be left aligned.
  - (b) **Record FIRST NAME** First five letters.
  - (c) **IDENTIFICATION NUMBER** Your UW WiseCard number.
  - (d) **Leave Special Codes ABCD** Leave these empty.
  - (e) **Record Special Codes EF as 21** This is exam version.
4. **Taking this exam indicates that you agree: to not write answers in large letters on exam paper for others to see, and to keep your scantron answers covered; to not view or use another student's work or use any unauthorized devices in any way; to not make any type of copy of any portion of this exam; and that you understand that being caught doing any of these actions, or other actions that may any student to see or submit work that is not wholly their own will result in automatic failure of the exam and possible failure of the course. Penalties are reported to the Deans Office for all involved.**

Parts	Number of Questions	Question Format	Possible Points
I	10	2 pt Simple Choice	20
II	17	3 pt Multiple Choice	51
III	1	1 pt Scantron Check	1
	28	Total	72

**Assumptions unless instructions explicitly state otherwise:**

addresses and integers are 4 bytes unless explicitly stated otherwise.  
 code questions are about C std=gnu99 and IA-32 on our Linux platform

**Reference: Powers of 2**

$$2^5 = 32, 2^6 = 64, 2^7 = 128, 2^8 = 256, 2^9 = 512, 2^{10} = 1024$$

$$2^{10} = K = \text{KiB}, 2^{20} = M = \text{MiB}, 2^{30} = G = \text{GiB}$$

$$2^A * 2^B = 2^{A+B}, 2^A / 2^B = 2^{A-B}$$

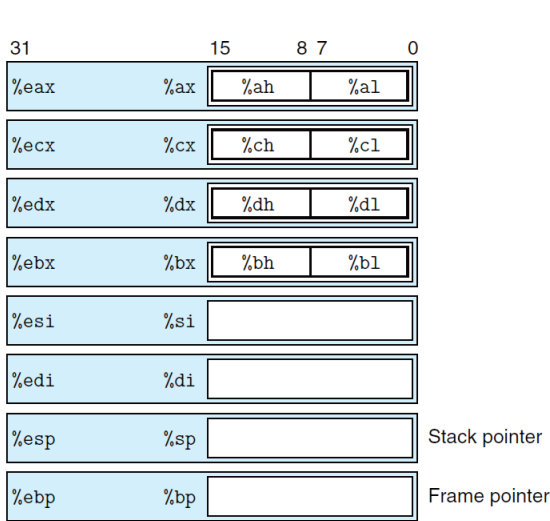
**Turn off and put away all notes and electronic devices and  
 wait for the proctor to signal the start of the exam.**

A

### END OF EXAM 2 (Special Codes EF-21)

Do not speak to classmates and keep your answers covered while waiting to submit.

### Reference: IA-32



Instruction	Effect	Description
MOV <i>S, D</i>	$D \leftarrow S$	Move
movb	Move byte	
movw	Move word	
movl	Move double word	
MOVS <i>S, D</i>	$D \leftarrow \text{SignExtend}(S)$	Move with sign extension
movsbw	Move sign-extended byte to word	
movsbl	Move sign-extended byte to double word	
movswl	Move sign-extended word to double word	
MOVZ <i>S, D</i>	$D \leftarrow \text{ZeroExtend}(S)$	Move with zero extension
movzbw	Move zero-extended byte to word	
movzbl	Move zero-extended byte to double word	
movzwl	Move zero-extended word to double word	
pushl <i>S</i>	$R[\%esp] \leftarrow R[\%esp] - 4;$ $M[R[\%esp]] \leftarrow S$	Push double word
popl <i>D</i>	$D \leftarrow M[R[\%esp]];$ $R[\%esp] \leftarrow R[\%esp] + 4$	Pop double word

Type	Form	Operand value	Name
Immediate	$\$Imm$	$Imm$	Immediate
Register	$E_a$	$R[E_a]$	Register
Memory	$Imm$	$M[Imm]$	Absolute
Memory	$(E_a)$	$M[R[E_a]]$	Indirect
Memory	$Imm(E_b)$	$M[Imm + R[E_b]]$	Base + displacement
Memory	$(E_b, E_i)$	$M[R[E_b] + R[E_i]]$	Indexed
Memory	$Imm(E_b, E_i)$	$M[Imm + R[E_b] + R[E_i]]$	Indexed
Memory	$(, E_i, s)$	$M[R[E_i] \cdot s]$	Scaled indexed
Memory	$Imm(, E_i, s)$	$M[Imm + R[E_i] \cdot s]$	Scaled indexed
Memory	$(E_b, E_i, s)$	$M[R[E_b] + R[E_i] \cdot s]$	Scaled indexed
Memory	$Imm(E_b, E_i, s)$	$M[Imm + R[E_b] + R[E_i] \cdot s]$	Scaled indexed

Figure 3.3 Operand forms. Operands can denote immediate (constant) values, register values, or values from memory. The scaling factor *s* must be either 1, 2, 4, or 8.

Instruction	Effect	Description
leal <i>S, D</i>	$D \leftarrow \&S$	Load effective address
INC <i>D</i>	$D \leftarrow D + 1$	Increment
DEC <i>D</i>	$D \leftarrow D - 1$	Decrement
NEG <i>D</i>	$D \leftarrow -D$	Negate
NOT <i>D</i>	$D \leftarrow \sim D$	Complement
ADD <i>S, D</i>	$D \leftarrow D + S$	Add
SUB <i>S, D</i>	$D \leftarrow D - S$	Subtract
IMUL <i>S, D</i>	$D \leftarrow D * S$	Multiply
XOR <i>S, D</i>	$D \leftarrow D \wedge S$	Exclusive-or
OR <i>S, D</i>	$D \leftarrow D   S$	Or
AND <i>S, D</i>	$D \leftarrow D \& S$	And
SAL <i>k, D</i>	$D \leftarrow D \ll k$	Left shift
SHL <i>k, D</i>	$D \leftarrow D \ll k$	Left shift (same as SAL)
SAR <i>k, D</i>	$D \leftarrow D \gg_A k$	Arithmetic right shift
SHR <i>k, D</i>	$D \leftarrow D \gg_L k$	Logical right shift