

EXAM 1

CIS 310

summer, 2021

Your Name Demetrius Johnson

****notice: I answered the questions on paper, scanned them, and took screenshots of them and pasted them into this document; my printer is out of ink at the moment, so I figured out a way to do this and make it still neat and easy to grade. Thank you(:**

Upload your answers on CANVAS by the midnight 6/6/21 (Sun). Write legibly.

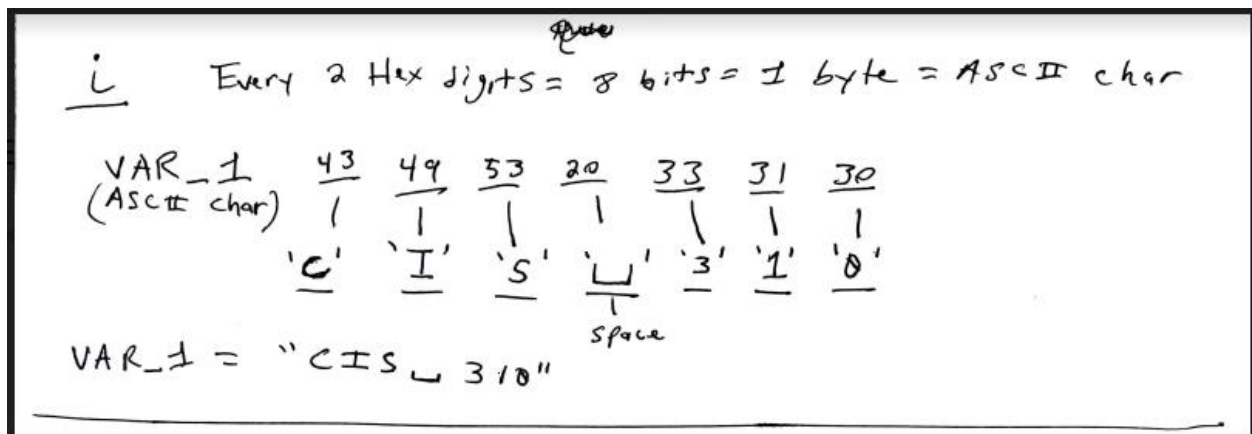
1. Decode (or interpret) the following as indicated. Use the ASC II table at the end of the exam if necessary.

VAR_1 43495320333130

VAR_2 E9

VAR_3 42F70000

- i. VAR_1 as ASCII characters (5 points)



ii. VAR_2 as a signed integer in one byte (show all work) (10 points)

Q VAR_2 E9 $\rightarrow (E9)_{16} = 8 \text{ bits} = 1 \text{ Hex digit}$
(Signed int, 8 bits)

$$E = (14)_{10} = 1110$$

$$9 = (9)_{10} = 1001$$

$$\begin{array}{l} \frac{14}{2} = 7 \quad R=0 \\ \frac{7}{2} = 3 \quad R=1 \\ \frac{3}{2} = 1 \quad R=1 \\ \frac{1}{2} = 0 \quad R=1 \end{array} \left. \vphantom{\begin{array}{l} \frac{14}{2} \\ \frac{7}{2} \\ \frac{3}{2} \\ \frac{1}{2} \end{array}} \right\} \begin{array}{l} 3^1, 2^1, 1^0 \\ (1110)_2 \end{array}$$

$$\begin{array}{l} \frac{9}{2} = 4 \quad R=1 \\ \frac{4}{2} = 2 \quad R=0 \\ \frac{2}{2} = 1 \quad R=0 \\ \frac{1}{2} = 0 \quad R=1 \end{array} \left. \vphantom{\begin{array}{l} \frac{9}{2} \\ \frac{4}{2} \\ \frac{2}{2} \\ \frac{1}{2} \end{array}} \right\} \begin{array}{l} 1^1, 0^1, 0^0, 1^0 \\ (1001)_2 \end{array}$$

Solution:

$(E9)_{16}$ Signed int
is $(-23)_{10}$

$$(E9)_{16} = (11101001)_2$$

The first bit is the
Sign bit = 1 = true = negative
So we know our value is a
negative integer.

2's Complement is used to
represent negative integers; thus
subtract 1, and flip bits:

$$11101001 - 1 = 11101000$$

$$\text{Flip} = 00010111$$

$$\begin{array}{l} \text{Convert to decimal: } 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\ = 16 + 4 + 2 + 1 = 23 \end{array}$$

but value is negative

So $(E9)_{16}$ as signed int = $\boxed{-23}$

iii. VAR_3 as An IEEE Floating point number in 4 bytes (show all work) (20 points).


iii VAR_3 = 42F70000 = 4 bytes or 32 bits
(IEEE FP#) $x = (-1)^S \cdot 2^E \cdot 1.F$

- Use shortcut: Every 4 bits = 1 Hex digit.

$\cdot (42F70000)_{16} =$

$\begin{array}{c} \text{4} \\ | \\ \cancel{0010} \\ 0100 \end{array}$ $\frac{2}{|}$ $\frac{\text{F}}{|}$ $\frac{7}{|}$ $\frac{0\ 0\ 0\ 0}{|}$

0010 1111 0111 $\underline{0000}$ $\underline{0000}$ $\underline{0000}$ $\underline{0000}$

• $(42F70000)_{16} =$ 

$S = 0 = \text{Positive}$

$$\text{b) } E = 10000101 = \frac{7654}{1000} \frac{3210}{0101} = 1 \times 2^7 + 1 \times 2^2 + 1 \times 2^0 = 128 + 4 + 1 = (133)_{10}$$

$$F = \underline{111011100000000000000000}$$

unbiased $E = (133)_{10} - (127)_{10} = (6)_{10}$

$$\cdot (-1)^0 * 2^6 * 1.1110111 = (1111011.1)_2$$

$$1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^1 + 1 \times 1^0$$
$$= 64 + 32 + 16 + 8 + 2 + 1 = (123)_{10}$$

$$(0.1)_2 = 0.1 \times 2 =$$

$$(0.1)_2 = 1 \cdot \frac{1}{2^1} = 1 \cdot \frac{1}{2} = \frac{1}{2} = (0.5)_{10}$$

Final answer: $(42F70000)_{16}$ IEEE FP = 123.5

2. Figure out the following FLAGS after the execution of the following instruction:
Z,S,O,C.

Assume data registers AL has 00101011 and BL 00011010.

ADD AL, BL (10 points)

Show the contents of AL after the execution of the instruction and figure out Z, S, O and C.

②

Add AL BL

Destination register Source register

$$\begin{array}{r} \text{AL} = \quad 00101011 \\ + \text{BL} = \quad 00011010 \\ \hline \quad 01000101 \end{array}$$

New Contents of AL ~~BL~~ \rightarrow $= 01000101$

Zero Flag = 0 (dest $\neq 0$)

Sign Flag = 0 (sign = +)

Carry Flag = 0 (Dest value in range for unsigned int)

overflow flag = 0 (value in range for signed int)

3. Draw the memory map of the following data: (put one byte in each cell) (25 points).

*note that the _ character in my map refers to the space ' ' character.

.DATA

First BYTE 12, 48

Name BYTE 'My name is Danny'

Num1 REAL4 123.5

(3)

	Memory Addr. (32-bits)	1 byte
First	0001:	12d
	0002:	48d
NAME	0003:	'M'
	0004:	'Y'
	0005:	'I'
	0006:	'N'
	0007:	'A'
	0008:	'M'
	0009:	'E'
	000A:	'L'
	000B:	'I'
	000C:	'S'
	000D:	'L'
	000E:	'D'
Num1	000F:	'A'
	0010:	'N'
	0011:	'N'
	0012:	'Y'
	0013:	00h
	0014:	00h
	0015:	F7h
	0016:	42h

* I accidentally started at memory Addr. 0001h, but meant to start at (0000)hex.

Little Endian Order

* All data types larger than a Byte store their individual bytes in reverse order, starting at LSB, at first (lowest) memory address.

32 bits = 4 bytes encoded Real

// Real4 > Byte; store in memory using

$(123.5)_{10}$ = The IEEE number evaluated in question 1-iii.

$(123.5)_{10} = (42F70000)_{16}$

$\begin{matrix} 1 & 2 & 3 & 4 \\ \text{byte} & & & \end{matrix}$

Little Endian order:

00 00 F7 42

Total bytes of memory on stack being used
= 22 bytes

4. Write program segment in X86 assembly language to compute 5×4 (10 points).

- I wrote my program in visual studio and ran it to make sure it worked and it did!(!)

```
;CIS-350 SUMMER 2021 WITH PROFESSOR DAVID YOON
```

```
;Name: Demetrius Johnson
```

```
;Date: 6-06-21
```

```
;Program Description: EXAM I Question 4: write a program that computes  $5 \times 4$ 
```

```
.386
```

```
.model flat,stdcall
```

```
.stack 4096
```

```
ExitProcess proto,dwExitCode:dword
```

```
.DATA
```

```
.CODE
```

```
main proc
```

```
mov al, 5d      ;move 5 into al 8-bit register 05h = 5d
```

```
mov bl, 4d      ;move 4 into bl 8-bit register 04h = 4d
```

```
mul bl          ;multiply al * bl --> it will be stored in the 16-bit AX register 0014h =20d
```

```
invoke ExitProcess,0
```

```
main endp
```

```
end main
```

Table 1.1 The ASCII character code

		0 000	1 001	2 010	3 011	4 100	5 101	6 110	7 111
0	0000	NUL	DLC	SP	0	@	P	'	p
1	0001	SOH	DC1	!	1	A	Q	a	q
2	0010	STX	DC2	"	2	B	R	b	r
3	0011	ETX	DC3	#	3	C	S	c	s
4	0100	EOT	DC4	\$	4	D	T	d	t
5	0101	ENQ	NAK	%	5	E	U	e	u
6	0110	ACK	SYN	&	6	F	V	f	v
7	0111	BEL	ETB	'	7	G	W	g	w
8	1000	BS	CAN	(8	H	X	h	x
9	1001	HT	EM)	9	I	Y	i	y
A	1010	LF	SUB	*	:	J	Z	j	z
B	1011	VT	ESC	+	;	K	[k	{
C	1100	FF	FS	,	<	L	\	l	
D	1101	CR	GS	-	=	M]	m	}
E	1110	SO	RS	.	>	N	^	n	~
F	1111	SI	US	/	?	O	_	o	DEL