CSCI 2510 Computer Organization 2020-21

Assignment 1

Deadline: October 6, 2020 (TUE) 14:30pm

Submission Notes:

(1) For each of the following written exercises (*Questions 1~3*), please show your steps and explain in detail when needed to receive full credit.

(2) Submit two files named **assignment1.pdf** (for *Question 1~3*) and **info.asm** (for *Programming Exercise*) to Blackboard Assignment Collection Box before the deadline (14:30pm on Oct 6).

(3) Late submission is not acceptable.

Question 1 (10 pts)

Describe the relationship among high-level programming language (e.g. C/C++), assembly language (e.g., MASM), and machine language (or machine code).

Question 2 (30 pts)

Consider a 32-bit word $\texttt{(B855486B)}_{16}$:

(a) What is it if interpreted as a string of characters (according to the below extended ASCII table)?

ASCII control characters			ASCII printable characters						Extended ASCII characters							
00	NULL	(Null character)	32	space	64	@	96	`	128	Ç	160	á	192	└	224	Ó
01	SOH	(Start of Header)	33	!	65	A	97	a	129	ü	161	í	193	┴	225	ß
02	STX	(Start of Text)	34	"	66	B	98	b	130	é	162	ó	194	┬	226	Ô
03	ETX	(End of Text)	35	#	67	C	99	c	131	â	163	ú	195	├	227	Ò
04	EOT	(End of Trans.)	36	$	68	D	100	d	132	ä	164	ñ	196	─	228	õ
05	ENQ	(Enquiry)	37	%	69	E	101	e	133	à	165	Ñ	197	┼	229	Õ
06	ACK	(Acknowledgement)	38	&	70	F	102	f	134	å	166	ª	198	ã	230	µ
07	BEL	(Bell)	39	'	71	G	103	g	135	ç	167	º	199	Ã	231	þ
08	BS	(Backspace)	40	(	72	H	104	h	136	ê	168	¿	200	╚	232	Þ
09	HT	(Horizontal Tab)	41	)	73	I	105	i	137	ë	169	®	201	╔	233	Ú
10	LF	(Line feed)	42	*	74	J	106	j	138	è	170	¬	202	╩	234	Û
11	VT	(Vertical Tab)	43	+	75	K	107	k	139	ï	171	½	203	╦	235	Ù
12	FF	(Form feed)	44	,	76	L	108	l	140	î	172	¼	204	╠	236	ý
13	CR	(Carriage return)	45	-	77	M	109	m	141	ì	173	¡	205	═	237	Ý
14	SO	(Shift Out)	46	.	78	N	110	n	142	Ä	174	«	206	╬	238	¯
15	SI	(Shift In)	47	/	79	O	111	o	143	Å	175	»	207	¤	239	´
16	DLE	(Data link escape)	48	0	80	P	112	p	144	É	176	░	208	ð	240	≡
17	DC1	(Device control 1)	49	1	81	Q	113	q	145	æ	177	▒	209	Ð	241	±
18	DC2	(Device control 2)	50	2	82	R	114	r	146	Æ	178	▓	210	Ê	242	‗
19	DC3	(Device control 3)	51	3	83	S	115	s	147	ô	179	│	211	Ë	243	¾
20	DC4	(Device control 4)	52	4	84	T	116	t	148	ö	180	┤	212	È	244	¶
21	NAK	(Negative acknowl.)	53	5	85	U	117	u	149	ò	181	Á	213	ı	245	§
22	SYN	(Synchronous idle)	54	6	86	V	118	v	150	û	182	Â	214	Í	246	÷
23	ETB	(End of trans. block)	55	7	87	W	119	w	151	ù	183	À	215	Î	247	¸
24	CAN	(Cancel)	56	8	88	X	120	x	152	ÿ	184	©	216	Ï	248	°
25	EM	(End of medium)	57	9	89	Y	121	y	153	Ö	185	╣	217	┘	249	¨
26	SUB	(Substitute)	58	:	90	Z	122	z	154	Ü	186	║	218	┌	250	·
27	ESC	(Escape)	59	;	91	[	123	{	155	ø	187	╗	219	█	251	¹
28	FS	(File separator)	60	<	92	\	124	|	156	£	188	╝	220	▄	252	³
29	GS	(Group separator)	61	=	93	]	125	}	157	Ø	189	¢	221	¦	253	²
30	RS	(Record separator)	62	>	94	^	126	~	158	×	190	¥	222	Ì	254	■
31	US	(Unit separator)	63	?	95	_			159	ƒ	191	┐	223	▀	255	nbsp
127	DEL	(Delete)														

(https://theasciicode.com.ar/)

(b) What is its value in decimal if interpreted as an unsigned integer?
(c) What is its value in decimal if interpreted as a signed integer using sign-and-magnitude?
(d) What is its value in decimal if interpreted as a signed integer using 1's-complement?
(e) What is its value in decimal if interpreted as a signed integer using 2's-complement?
(f) What is its value in decimal if interpreted as a floating-point number using IEEE Standard 754 Single Precision?

Question 3 (30 pts)

Consider a computer system of word size 32 bits and has a main memory system of 3 GB.

(a) How many bits, bytes, and words are there in the memory system?
(b) If the system is byte addressable, what is the minimum number of required bits for memory addresses?
(c) Suppose a 32-bit number **`(6A738C9E)`**$_{16}$ is stored at word address 200, and a string of characters "**`CSCI2510`**" is stored at word address 204. Please fill the contents of the memory in the following forms when 1) big-endian system and 2) little-endian system are adopted, respectively. *(Note: An N-character string should not be treated as one large multi-byte value, but rather as N single character values. That is, the first character of the string has the smallest byte address, while the last character has the largest byte address.)*

Big endian

	+0	+1	+2	+3
200				
204				
208				

Little endian

	+3	+2	+1	+0
200				
204				
208				

Programming Exercise (30 pts)

Write a complete MASM IA-32 assembly program named **info.asm** to print your name, student ID, college, and major to the terminal using the "**`crt_printf`**" print function. *(Hint: Please refer to Tutorial 2 for more details.)*