

以下題目均按學號規則推算出指定的題數來做，學號為奇數者做奇數題，偶數者做偶數題，做錯題目則該題不計分，共 99 分，1 分奉送！

名詞解釋：每題 3 分，做指定的 6 題，共 18 分。

1. 目的檔 (object file) :
2. objdump
3. 動態連結 (Dynamic Linking)
4. 連結器 (Linker)
5. 組合語言 (Assembly Language)
6. 假指令 (Pseudo Instruction)
7. make
8. gcc
9. 暫存器 (register)
10. 算術邏輯單元 (ALU)
11. 程式計數器 (PC, Program Counter)
12. 連結暫存器 (LR, Link Register)

簡答題：每題 5 分，做指定的 4 題，共 20 分。

- |  |  |
|--|--|
| <ol style="list-style-type: none"><li>1. 請畫出一顆 CPU 的架構示意圖，並說明其中各個元件的功用？</li><li>2. 請畫出電腦的馮紐曼架構，並說明其中各個元件的功用？</li></ol>                                 | <ol style="list-style-type: none"><li>3. 請說明組譯器第一階段的功能？</li><li>4. 請說明組譯器第二階段的功能？</li></ol>                          |
| <ol style="list-style-type: none"><li>5. 請寫出 XOR R1, R2, R3 的機器碼 (16 進位寫法)，並說明其編碼的原理？</li><li>6. 請寫出 SHL R1, R2, 4 的機器碼 (16 進位寫法)，並說明其編碼的原理？</li></ol> | <ol style="list-style-type: none"><li>7. 請舉例說明 CPU0 中跳躍指令 JMP 的運作原理。</li><li>8. 請舉例說明 CPU0 中儲存指令 ST 的運作原理。</li></ol> |

考生姓名：

學號：

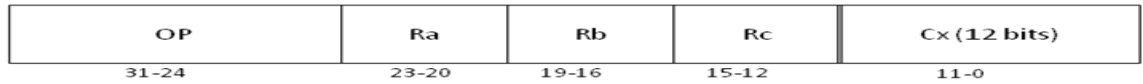
程式題 (請以 CPU0 的組合語言撰寫)：每題 10 分，做指定的 4 題，共 40 分。

<ol style="list-style-type: none"><li>1. 請寫出一個組合語言程式，可以將變數 X, Y 的相加後放到變數 Z 中。</li><li>2. 請寫出一個組合語言程式，可以將變數 X 的內容加 10 後放回 X 中。</li> <li>5. 請寫出一個組合語言程式，說明如何透過暫存器傳遞參數。</li><li>6. 請寫出一個「組合語言副程式」，可以設定 R1 為 <math>3^{R2}</math>。</li></ol>	<ol style="list-style-type: none"><li>3. 請寫出一個組合語言程式，可以取得變數 X, Y 中較小的值放入變數 min 中。</li><li>4. 請寫出一個組合語言程式，可以取得變數 X, Y 中較大的值放入變數 max 中。</li> <li>7. 請寫出一個組合語言程式，說明如何透過堆疊傳遞參數。</li><li>8. 請寫出一個組合語言程式，說明如何透過堆疊傳遞參數。</li></ol>
---	--

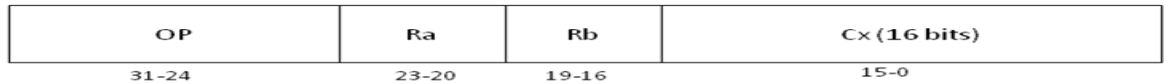
請寫出下列程式的位址欄與機器碼 (每列 3 分，位址欄 1 分，機器碼 2 分，做指定的 7 列，共 21 分)

題號	位址欄	組合語言	機器碼
1.		LDI R1, 1	
2.		LD R2, aptr	
3.		LDI R4, 1	
4.		FOR: LDB R3, [R2]	
5.		CMP R3, R0	
6.		JEQ EXIT	
7.		MUL R4, R4, R3	
8.		ADD R2, R2, R1	
9.		JMP FOR	
10.		EXIT: ST R4, f	
11.		RET	
12.		a: BYTE 3, 2, 7, 6, 8, 5, 0	
13.		f: WORD 0	
14.		aptr: WORD a	

## A 型



## L 型



## J 型



類型	格式	指令	OP	說明	語法	語意
載入 儲存	L	LD	00	載入 word	LD Ra, [Rb+Cx]	$Ra \leftarrow [Rb + Cx]$
	L	ST	01	儲存 word	ST Ra, [Rb+ Cx]	$Ra \rightarrow [Rb + Cx]$
	L	LDB	02	載入 byte	LDB Ra, [Rb+ Cx]	$Ra \leftarrow (\text{byte})[Rb + Cx]$
	L	STB	03	儲存 byte	STB Ra, [Rb+ Cx]	$Ra \rightarrow (\text{byte})[Rb + Cx]$
	A	LDR	04	LD 的暫存器版	LDR Ra, [Rb+Rc]	$Ra \rightarrow (\text{byte})[Rb + Rc]$
	A	STR	05	ST 的暫存器版	STR Ra, [Rb+Rc]	$Ra \rightarrow [Rb + Rc]$
	A	LBR	06	LDB 的暫存器版	LBR Ra, [Rb+Rc]	$Ra \leftarrow (\text{byte})[Rb + Rc]$
	A	SBR	07	STB 的暫存器版	SBR Ra, [Rb+Rc]	$Ra \rightarrow (\text{byte})[Rb + Rc]$
	L	LDI	08	立即載入	LDI Ra, Rb+Cx	$Ra \leftarrow Rb + Cx$
運算 指令	A	CMP	10	比較	CMP Ra, Rb	$SW \leftarrow Ra \geq Rb$
	A	MOV	12	移動	MOV Ra, Rb	$Ra \leftarrow Rb$
	A	ADD	13	加法	ADD Ra, Rb, Rc	$Ra \leftarrow Rb + Rc$
	A	SUB	14	減法	SUB Ra, Rb, Rc	$Ra \leftarrow Rb - Rc$
	A	MUL	15	乘法	MUL Ra, Rb, Rc	$Ra \leftarrow Rb * Rc$
	A	DIV	16	除法	DIV Ra, Rb, Rc	$Ra \leftarrow Rb / Rc$
	A	AND	18	邏輯 AND	AND Ra, Rb, Rc	$Ra \leftarrow Rb \text{ and } Rc$
	A	OR	19	邏輯 OR	OR Ra, Rb, Rc	$Ra \leftarrow Rb \text{ or } Rc$
	A	XOR	1A	邏輯 XOR	XOR Ra, Rb, Rc	$Ra \leftarrow Rb \text{ xor } Rc$
	A	ROL	1C	向左旋轉	ROL Ra, Rb, Cx	$Ra \leftarrow Rb \text{ rol } Cx$
	A	ROR	1D	向右旋轉	ROR Ra, Rb, Cx	$Ra \leftarrow Rb \text{ ror } Cx$
	A	SHL	1E	向左移位	SHL Ra, Rb, Cx	$Ra \leftarrow Rb \ll Cx$
	A	SHR	1F	向右移位	SHR Ra, Rb, Cx	$Ra \leftarrow Rb \gg Cx$
跳躍 指令	J	JEQ	20	跳躍 (相等)	JEQ Cx	if SW(=) $PC \leftarrow PC + Cx$
	J	JNE	21	跳躍 (不相等)	JNE Cx	if SW(!=) $PC \leftarrow PC + Cx$
	J	JLT	22	跳躍 (<)	JLT Cx	if SW(<) $PC \leftarrow PC + Cx$
	J	JGT	23	跳躍 (>)	JGT Cx	If SW(>) $PC \leftarrow PC + Cx$
	J	JLE	24	跳躍 (<=)	JLE Cx	if SW(<=) $PC \leftarrow PC + Cx$
	J	JGE	25	跳躍 (>=)	JGE Cx	If SW(>=) $PC \leftarrow PC + Cx$
	J	JMP	26	跳躍 (無條件)	JMP Cx	$PC \leftarrow PC + Cx$
	J	SWI	2A	軟體中斷	SWI Cx	$LR \leftarrow PC; PC \leftarrow Cx$
	J	CALL	2B	跳到副程式	CALL Cx	$LR \leftarrow PC; PC \leftarrow PC + Cx$
J	RET	2C	返回	RET	$PC \leftarrow LR$	
堆疊 指令	A	PUSH	30	推入 word	PUSH Ra	$SP -= 4; [SP] = Ra;$
	A	POP	31	彈出 word	POP Ra	$Ra = [SP]; SP += 4;$
	A	PUSHB	32	推入 byte	PUSHB Ra	$SP--; [SP] = Ra; (\text{byte})$
	A	POPB	33	彈出 byte	POPB Ra	$Ra = [SP]; SP++; (\text{byte})$