

Final Examination
CS2422 Assembly Language and System Programming
January 8, 2008

1. (8%) Consider the following macro definition.

```
mPutchar MACRO char
    push eax
    mov al,char
    call WriteChar
    pop eax
ENDM
```

Write down the expanded code if the macro is invoked as follows.

```
.code
mPutchar.
```

2. (15%) Suppose **GETC** is a subroutine to read one character from the input device.

- (a) (5%) Please state what the following program is doing.
 (b) (10%) Please explain what each line of the program is doing.

```
PROG START      0
      CLEAR     A
      LDX       #128
LOOP JSUB       GETC
      RMO      A,S
      SHIFTL   S,4
      JSUB     GETC
      ADDR    S,A
      STCH    0,X
      TIXR   X
      J      LOOP
```

3. (25%) Suppose the following program is assembled by a SIC assembler.

Loc	Source	statement	Object code
1000	COPY	START 1000	
1000	FIRST	STL RETADR	141033
1003	CLOOP	JSUB IOREC	481439
1006		LDA LENGTH	001036
1009		COMP ZERO	281030
100C		JEQ ENDFIL	301015
100F		JSUB IOREC	481439
1012		J CLOOP	3C1003
1015	ENDFIL	LDA EOF	00102A
1018		STA BUFFER	0C1039
101B		LDA THREE	00102D
101E		STA LENGTH	0C1036
1021		JSUB IOREC	481439
1024		LDL RETADR	081033
1027		RSUB	4C0000
102A	EOF	BYTE C'EOF'	454F46
102D	THREE	WORD 3	000003
1030	ZERO	WORD 0	000000
1033	RETADR	RESW 1	
1036	LENGTH	RESW 1	
1039	BUFFER	RESB 1024	
1439	IOREC	...	

- (a) (5%) Consider a SIC/XE assembler that assembles the same code. Do the locations of the statements need to be changed? Give your reasons. Note that **BUFFER** has only 1024 bytes.
- (b) (15%) Write the resultant object code. The PC-relative instruction format is shown below.

	n	i	x	b	p	e	
opcode	1	1		0	1		displacement

- (c) (5%) Does any of the object code in (b) need to be changed if the above program is relocated to start from address 2000? If yes, which lines should be changed? If no, why?
4. (7%) Which of the following is not part of a language's calling convention? Give your reasons.
- which registers must be preserved by procedures
 - the order in which arguments are passed to procedures
 - how the stack pointer is restored after a procedure call
 - the size factor when decrementing the stack pointer
5. (10%) Explain the difference between a hardware interrupt and a software interrupt.
6. (35%) The following three object files are for a program. In this question, we will consider a linking loader to link and load these object files.

Object file A:

```
H PROGA 000000 000063
D LISTA 000040 ENDA 000054
R LISTB ENDB LISTC ENDC
...
T 000020 0A 03201D77100004050014
...
T 000054 0F 000014FFFFFF600003F000014FFFFC0
M 000024 05 +LISTB
M 000054 06 +LISTC
M 000057 06 +ENDC
M 000057 06 -LISTC
M 00005A 06 +ENDC
M 00005A 06 -LISTC
M 00005A 06 +PROGA
M 00005D 06 -ENDA
M 00005D 06 +LISTB
M 000060 06 +LISTB
M 000060 06 -PROGA
E
```

Object file B:

```
H PROGB 000000 00007F
D LISTB 000060 ENDB 000070
R LISTA ENDA LISTC ENDC
...
T 000036 0B 0310000077202705100000
...
T 000070 0F 000000FFFFFF6FFFFFFF0000060
M 000037 05 +LISTA
M 00003E 05 +ENDA
M 00003E 05 -LISTA
M 000070 06 +ENDA
M 000070 06 -LISTA
M 000070 06 +LISTC
M 000073 06 +ENDC
M 000073 06 -LISTC
M 000076 06 +ENDC
M 000076 06 -LISTC
```

```

M 000076 06 +LISTA
M 000079 06 +ENDA
M 000079 06 -LISTA
M 00007C 06 +PROGB
M 00007C 06 -LISTA
E

```

Object file C:

```

H PROGC 000000 000051
D LISTC 000030 ENDC 000042
R LISTA ENDA LISTB ENDB
...
T 000018 0C 031000007710000405100000
...
T 000042 0F 000030000008000011000000000000
M 000019 05 +LISTA
M 00001D 05 +LISTB
M 000021 05 +ENDA
M 000021 05 -LISTA
M 000042 06 +ENDA
M 000042 06 -LISTA
M 000042 06 +PROGC
M 000048 06 +LISTA
M 00004B 06 +ENDA
M 00004B 06 -LISTA
M 00004B 06 -ENDB
M 00004B 06 +LISTB
M 00004E 06 +LISTB
M 00004E 06 -LISTA
E 000018

```

- (a) (10%) Suppose the start loading address of this program is 135A, and the object files are linked and loaded in the order of file B, file A and file C. During pass 1 of the linking loader, the following external symbol table ESTAB will be generated. Please fill out the blanks.

Control section	Symbol name	Address	Length
PROGB	LISTB	_____	_____
	ENDB	_____	_____
		_____	_____
PROGA	LISTA	_____	_____
	ENDA	_____	_____
		_____	_____
PROGC	LISTC	_____	_____
	ENDC	_____	_____
		_____	_____

- (b) (12%) Suppose the loading is just finished.
- (i) (6%) What is the corresponding instruction for the opcode at memory address 1397?
And please also write the whole instruction line corresponding to this instruction.
 - (ii) (6%) What is the corresponding instruction for the opcode at memory address 1454?
And please also write the whole assembly instruction line corresponding to this instruction.
- (c) (8%) After the loading, what are the values of the variables corresponding to memory addresses of 13D6 and 1433, respectively?
- (d) (5%) What is the starting execution address of the program?