National Sun Yat-Sen University
ASSEMBLY LANGUAGE AND MICROCOMPUTER
Midterm Exam
2:20-4:20PM Nov 11 2010

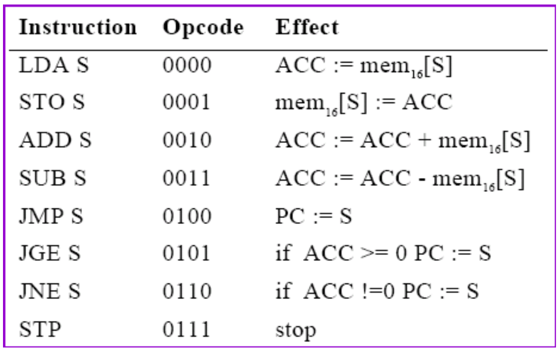
Name:__________________

Note: Although there are total 104 points for this exam, the maximum score you can get is 100 points.

1. Translate the following C-codes into the assembly codes based on the simple MU0 instruction set. In addition, translate the assembly code into the binary code. You should describe the initial contents of the memory when your program start running. ***(12 pts)***

```
int a, b,c;

if (a=b) c=a+b;
else c=a-b+127;
```



Instruction	Opcode	Effect
LDA S	0000	ACC := mem_{16}[S]
STO S	0001	mem_{16}[S] := ACC
ADD S	0010	ACC := ACC + mem_{16}[S]
SUB S	0011	ACC := ACC - mem_{16}[S]
JMP S	0100	PC := S
JGE S	0101	if ACC >= 0 PC := S
JNE S	0110	if ACC !=0 PC := S
STP	0111	stop

2. Specify a single ARM instruction which implements the following equation: ***(9 pts)***
   (a) r0 = 17*r2 + C (C:carry bit) (b) mem[r1+4]=$r2_{[7:0]}$, r1=r1+4;
   (c) r1 = 0xFFFFFFFF ;

3. Suppose r1=0xF0000001, r2=0xF0000000 and C=1, N=0, Z=0, V=0, find out the resulting **r1** value of the following instructions. You should also provide the resulting conditional code value (**C N Z**). ***(20 pts)***
   (a) ADC r1, r1, r2 (b) SBCS r1, r1, r2
   (c) MOV r1, r2, ROR #3 (d) BICEQS r1, r2, r2.

4. The contents of registers r0 to r6 need to be swapped according to the following description:
   - r0 moved into r3
   - r1 moved into r4
   - r2 moved into r6
   - r3 moved into r5
   - r4 moved into r0
   - r5 moved into r1
   - r6 moved into r2

   Write a segment of code that uses the minimum number of full descending stack operations to carry this out, and hence requires no use of any other registers for temporary storage. ***(8 pts)***

5. Explain the following two popular addressing modes: (1) Register (2) Base plus index. You should provide an ARM instruction as an example for each mode. ***(8 pts)***

6. Answer for the following short questions: ***(10 pts)***

   (a) List two ARM exception types except software interrupt. Discuss how to invoke each of two listed exceptions. ***(6 pts)***

   (b) Explain what SPSR is and what it is used for. ***(4 pts)***

7. Find out if the following instruction format is correct or not. If incorrect, point out the problem. ***(15 pts)***

   (a) STMIA r5!, {r5, r4, r9}

   (b) ADD r3, r7, #258

   (c) STR pc, [r1, r0, LSL #2]

   (d) MOVS r4, r4, RRX #1

   (e) LDRS r7, [r3, #12]

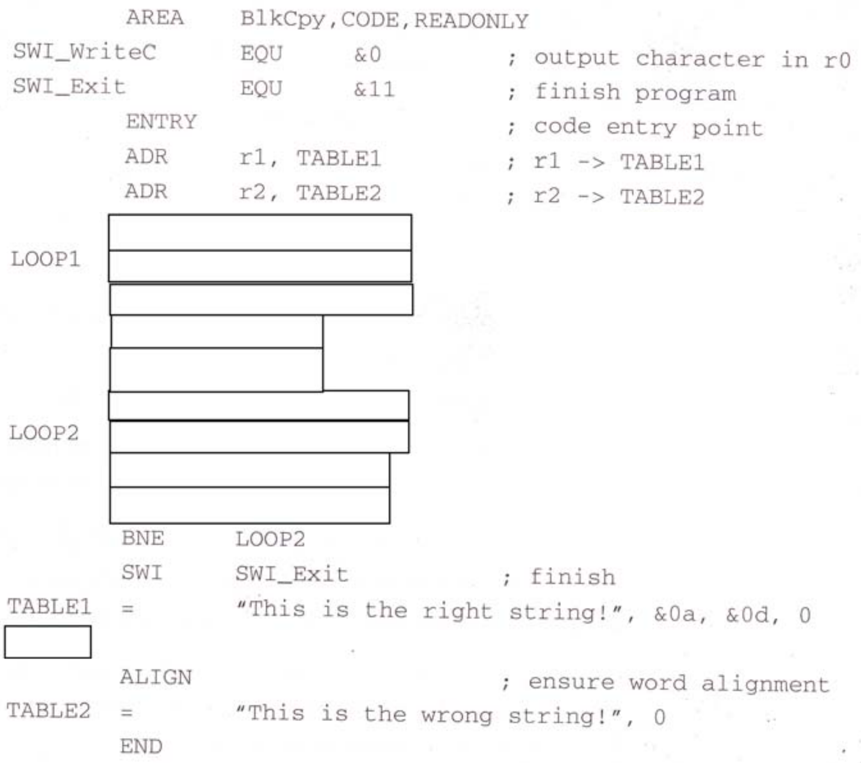
8. For the following C-expression: ***(12 pts)***

   if ((r0==r1+1) && (r2!=r3)) r4=4*r1-r3;

   (a) Write down the corresponding ARM code without using the conditional execution. (But you can still use conditional branches.)

   (b) Write down the corresponding ARM code using the minimum number of instructions.

9. Fill the flowing missing blanks such that this program can use word loads and stores to copy the string from **TABLE1** to **TABLE2**. This program will also print out the string copied using the SWI instruction. ***(10 pts)***



```
          AREA      BlkCpy,CODE,READONLY
SWI_WriteC          EQU      &0          ; output character in r0
SWI_Exit            EQU      &11         ; finish program
          ENTRY                          ; code entry point
          ADR       r1, TABLE1           ; r1 -> TABLE1
          ADR       r2, TABLE2           ; r2 -> TABLE2

LOOP1






LOOP2


          BNE       LOOP2
          SWI       SWI_Exit             ; finish
TABLE1    =         "This is the right string!", &0a, &0d, 0

          ALIGN                          ; ensure word alignment
TABLE2    =         "This is the wrong string!", 0
          END
```