

CSE2312-002/003, Fall 2014, Homework 2

Paper Copy Due in Class: September 16, 2014 (at 2pm for 002, at 3:30pm for 003)

The following problems are from Chapter 2 of the ARM Edition of the Patterson and Hennessy textbook (available on Blackboard as PDFs under Course Materials). In addition to the resources available in Chapter 2, you may find Appendices B1 and B2 of the ARM Edition useful for these problems (also available as PDFs on Blackboard).

The problems below respectively correspond to 2.1, 2.4, 2.5, and 2.7 in the book.

2.1. The following problems deal with translating from C to ARM. Assume that the variables g , h , i , and j are given and could be considered 32-bit integers as declared in a C program.

a.	$f = g + h + i + j;$
b.	$f = g + (h + 5);$

2.1.1 For the C statements above, what is the corresponding ARM assembly code? Use a minimal number of ARM assembly instructions.

2.1.2 For the C statements above, how many ARM assembly instructions are needed to perform the C statement?

2.1.3 If the variables f , g , h , j , and i have values 1, 2, 3, 4, and 5, respectively, what is the end value of f ?

The following problems deal with translating from ARM to C. Assume that the variables g , h , i , and j are given and could be considered 32-bit integers as declared in a C program.

a.	ADD f, g, h
b.	ADD $f, f, \#1$ ADD f, g, h

2.1.4 For the ARM statements above, what is a corresponding C statement?

2.1.5 If the variables f , g , h , and i have values 1, 2, 3, and 4, respectively, what is the end value of f ?

2.4. The following problems deal with translating from C to ARM. Assume that the variables f , g , h , i , and j are assigned to registers $r0$, $r1$, $r2$, $r3$, and $r4$, respectively. Assume that the base address of the arrays A and B are in registers $r6$ and $r7$, respectively.

a.	$f = g + h + B[4];$
b.	$f = g - A[B[4]];$

2.4.1 For the C statements above, what is the corresponding ARM assembly code?

2.4.2 For the C statements above, how many ARM assembly instructions are needed to perform the C statement?

2.4.3 For the C statements above, how many different registers are needed to carry out the C statement? The following problems deal with translating from ARM to C. Assume that the variables *f*, *g*, *h*, *i*, and *j* are assigned to registers *r0*, *r1*, *r2*, *r3*, and *r4*, respectively.

Assume that the base address of the arrays A and B are in registers *r6* and *r7*, respectively.

a.	<pre> ADD r0, r0, r1 add r0, r0, r2 add r0, r0, r3 add r0, r0, r4 </pre>
b.	<pre> LDR r0, [r6, #0x4] </pre> <p>NOTE: there was a typo, it should be as above, not LDR r0, [r6, #0x4]</p>

2.4.4 For the ARM assembly instructions above, what is the corresponding C statement?

2.4.5 For the ARM assembly instructions above, rewrite the assembly code to minimize the number of ARM instructions (if possible) needed to carry out the same function.

2.4.6 How many registers are needed to carry out the ARM assembly as written above? If you could rewrite the code above, what is the minimal number of registers needed?

2.5. In the following problems, we will be investigating memory operations in the context of an ARM processor. The table below shows the values of an array stored in memory.

a.	<table> <tr> <th>Address</th><th>Data</th></tr> <tr> <td>12</td><td>1</td></tr> <tr> <td>8</td><td>6</td></tr> <tr> <td>4</td><td>4</td></tr> <tr> <td>0</td><td>2</td></tr> </table>	Address	Data	12	1	8	6	4	4	0	2		
Address	Data												
12	1												
8	6												
4	4												
0	2												
b.	<table> <tr> <th>Address</th><th>Data</th></tr> <tr> <td>16</td><td>1</td></tr> <tr> <td>12</td><td>2</td></tr> <tr> <td>8</td><td>3</td></tr> <tr> <td>4</td><td>4</td></tr> <tr> <td>0</td><td>5</td></tr> </table>	Address	Data	16	1	12	2	8	3	4	4	0	5
Address	Data												
16	1												
12	2												
8	3												
4	4												
0	5												

2.5.1 For the memory locations in the table above, write C code to sort the data from lowest-to-highest, placing the lowest value in the smallest memory location shown in the figure. Assume that the data shown represents the C variable called Array, which is an array of type int. Assume that this particular machine is a byte-addressable machine and a word consists of 4 bytes.

2.5.2 For the memory locations in the table above, write ARM code to sort the data from lowest-to-highest, placing the lowest value in the smallest memory location. Use a minimum number of ARM instructions. Assume the base address of Array is stored in register *r6*.

2.5.3 To sort the array above, how many instructions are required for the ARM code? If you are not allowed to use the immediate field in *lw* and *sw* instructions, how many ARM instructions do you need?

The following problems explore the translation of hexadecimal numbers to other number formats.

a.	0x12345678
b.	0xbeadf00d

2.5.4 Translate the hexadecimal numbers above into decimal.

2.5.5 Show how the data in the table would be arranged in memory of a little-endian and a big-endian machine. Assume the data is stored starting at address 0.

2.7. The following problems explore number conversions from signed and unsigned binary number to decimal numbers.

a. 1010 1101 0001 0000 0000 0000 0000 0010_{two}

b. 1111 1111 1111 1111 1011 0011 0101 0011_{two}

2.7.1 For the patterns above, what base 10 number does it represent, assuming that it is a two's complement integer?

2.7.2 For the patterns above, what base 10 number does it represent, assuming that it is an unsigned integer?

2.7.3 For the patterns above, what hexadecimal number does it represent? The following problems explore number conversions from decimal to signed and unsigned binary numbers.

a. 2147483647_{ten}

b. 1000_{ten}

2.7.4 For the base ten numbers above, convert to 2's complement binary.

2.7.5 For the base ten numbers above, convert to 2's complement hexadecimal.

2.7.6 For the base ten numbers above, convert the negated values from the table to 2's complement hexadecimal.