**Question 1.a:**

**Ex. 2.4.1:** 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.

1. f = g + h + B[4];
2. f = g – A[B[4]];

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

**Answer 1.a:**

1. ADD r0, r1, r2 ; f = g + h

LDR r8, [r7, #16] ; temp r8 = B[4]

ADD r0, r0, r8 ; f += B[4]

1. LDR r8, [r7, #16] ; temp r8 = B[4]

LDR r8, [r6, r8] ; temp r8 = A[B[4]]

SUB r0, r1, r8 ; f = g - A[B[4]]

**Question 1.b:**

**Ex. 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 fi gure. 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.

|  |  |  |
| --- | --- | --- |
| a. | Address  12  8  4  0 | Data  1  6  4  2 |
| b. | Address  16  12  8  4  0 | Data  1  2  3  4  5 |

**Answer 1.b:**

1. void task\_2\_5\_1\_a(int \* Array)

{

int i, j;

for (i = 0; i < 3; i++)

{

for (j = 0; j < 12; j += 4)

{

/\* Assuming Array[x] is accessing data at Array with x bytes offset \*/

if (Array[j] > Array[j + 4])

{

int tmp = Array[j];

Array[j] = Array[j + 4];

Array[j + 4] = tmp;

}

}

}

}

1. void task\_2\_5\_1\_b(int \* Array)

{

int i, j;

for (i = 0; i < 3; i++)

{

for (j = 0; j < 16; j += 4)

{

/\* Assuming Array[x] is accessing data at Array with x bytes offset \*/

if (Array[j] > Array[j + 4])

{

int tmp = Array[j];

Array[j] = Array[j + 4];

Array[j + 4] = tmp;

}

}

}

}

**Question 1.e**:

**Ex. 2.13:** In the following problems, the data table contains the values for registers r3 and r4. You will be asked to perform several ARM logical operations on these registers.

1. r3 = 0x55555555, r4 = 0x12345678
2. r3 = 0xBEADFEED, r4 = 0xDEADFADE

**2.13.1** For the lines above, what is the value of r5 for the following sequence of instructions:

OR r5, r4, r3, LSL #4

**2.13.2** For the values in the table above, what is the value of r5 for the following sequence of instructions:

MVN r3, #1

AND r5, r3, r4, LSL #4

**2.13.3** For the lines above, what is the value of r5 for the following sequence of instructions:

MOV r5, 0xFFEF

AND r5, r5, r3, LSR #3

**Answer 1.e:**

**2.13.1**: There is no OR instruction, so assuming it’s ORR the answers will be:

1. 0x57755778
2. 0xfefffede

**2.13.2:**

1. 0x23456780
2. 0xeadfade0

**2.13.3:** Immediate value 0xFFEF can’t be loaded with the MOV instruction. Assuming it’s loaded with LDR answers will be:

1. 0x0000aaaa
2. 0x0000bfcd

**Question 1.f:**

**Ex 2.16:** For these problems, the table holds various binary values for register r0. Given the

value of r0, you will be asked to evaluate the outcome of different branches.

1. 1010 1101 0001 0000 0000 0000 0000 0010two = AD100002hex
2. 1111 1111 1111 1111 1111 1111 1111 1111two = FFFFFFFFhex

**2.16.1** Suppose that register r0 contains a value from above and r1 has the value

0011 1111 1111 1000 0000 0000 0000 0000two = 3FF80000hex

What is the value of r2 after the following instructions?

MOV r2, #0

CMP r0, r1

BGE ELSE

B DONE

ELSE: MOV r2, #2

DONE:

**2.16.2** Suppose that register r0 contains a value from above and r1 has the value

0011 1111 1111 1000 0000 0000 0000 0000two = 0x3FF80000hex

What is the value of r2 after the following instructions?

MOV r2, #0

CMP r0, r1

BLO ELSE

B DONE

ELSE: MOV r2, #2

DONE:

**2.16.3** Rewrite the above code using conditional instructions of ARM.

For these problems, the table holds various binary values for register r0. Given the value of r0, you will be asked to evaluate the outcome of different branches.

1. 0x00001000
2. 0x20001400

**2.16.4** Suppose that register r0 contains a value from above. What is the value of r2 after the following instructions?

MOV r2, #0

CMP r0, r0

BLT ELSE

B DONE

ELSE: ADD r2, r2, #2

DONE:

**2.16.5** Suppose that register r0 contains a value from above. What is the value of r2 after the following instructions?

MOV r2, #0

CMP r0, r0

BHI ELSE

B DONE

ELSE: ADD r2, r2, #2

DONE:

**Answer 1.f:**

**2.16.1**

1. 0
2. 0

**2.16.2**

1. 0
2. 0

**2.16.3**

1. Code for the 2.16.1:

MOV r2, #0

CMP r0, r1

MOVGE r2, #2

1. Code for the 2.16.2:

MOV r2, #0

CMP r0, r1

MOVLO r2, #2

**2.16.4**

1. 0
2. 0

**2.16.5**

1. 0
2. 0

**Question 1.g:**

**Ex 2.17.4**:

1. LOOP: CMP r1, #0

BGT ELSE

B DONE

ELSE: ADD r2, r2, #2

SUB r1, r1, #1

B LOOP

DONE:

1. LOOP: MOV r4, 0xA

MOV r1, 0xA

LOOP2: ADD r2, r2, #2

SUB r1, r1, #1

CMP r1, #0

BNE LOOP2

SUB r4, r4, #1

CMP r4, #0

BNE LOOP

DONE:

**Answer 1.g:**

1. 20
2. 200.

**Question 1.h:  
Ex 2.18.2:**

For these problems, the table holds some C code. You will be asked to evaluate these C code statements in ARM assembly code.

1. for(i=0; i<10; i++)  
    a += b;
2. while (a < 10) {  
    D[a] = b + a;  
    a += 1;  
   }

For the table above, translate the C code to ARM assembly code. Use a minimum number of instructions. Assume that the value a, b, i, are in registers r0, r1, r2 respectively. Also, assume that register r3 holds the base address of the array D.

**Answer 1.h:**

1. MOV r2, #0 ; i = 0  
   LOOP: CMP r2, #10 ; i < 10 condition  
    BHS DONE  
    ADD r0, r0, r1 ; a += b  
    ADD r2, r2, #1 ; i++  
    B LOOP  
   DONE:
2. LOOP: CMP r0, #10 ; a < 10 condition

BHS DONE

ADD r4, r1, r0 ; b + a

STR r4, [r3, r0, LSL #2] ; D[a] = b + a

ADD r0, r0, #1 ; a += 1

B LOOP

DONE: