```
ï»¿UCR EE/CS 120B Lab 22 − Introduction to C and RIMS
Name: Ariana DeJaco adeja001@ucr.edu
Partner: Joshua DeForest−Williams jdefo002@ucr.edu


Contributions: Ariana (50%), Joshua (50%). We both put in 50/50 participation in
 this lab and helped each other in debugging pieces of code that we may have str
uggled with.


Pre−lab −− no prelab for this lab


Introductory material −− no problems, everything worked smoothly. The lab itself
 was long, however.

Day 1

Exercise 1 −− No issues.
Exercise 2 −− No issues
Exercise 3 −− No issues
Exercise 4 −− No issues
Exercise 5 −− No issues

Day 2
Exercise 1 −− No issues.
Exercise 2 −− No issues


```

```
1  /*      adeja001_lab0_part1 − 4/1/13
2   *      Name & E−mail: Ariana Dejaco adeja001@ucr.edu
3   *      CS Login: Adeja001
4   *      Partner(s) Name & E−mail: Josuha DeForest−Williams jdefo002@ucr.edu
5   *      Lab Section: 22
6   *      Assignment: Lab #1  Exercise # Chapter 1 Try 1
7   *      Exercise Description:
8   *      Run the above program. The programâM−^@M−^Ys behavior should appear iden
    tical to the earlier program having just âM−^@M−^\B = A;âM−^@M−^].
9   *      I acknowledge all content contained herein, excluding template or exampl
    e
10  *      code, is my own original work.
11  */
12 #include "RIMS.h"
13
14 void main() {
15 while (1) {
16 B0 = A0;
17 B1 = A1;
18 B2 = A2;
19 B3 = A3;
20 B4 = A4;
21 B5 = A5;
22 B6 = A6;
23 B7 = A7;
24 }
25 }
```

```
1  /*      adeja001_lab0_part2.c − 4/1/13
2   *      Name & E−mail: Ariana Dejaco adeja001@ucr.edu
3   *      CS Login: adeja001
4   *      Partner(s) Name & E−mail: Josuha DeForest−Williams jdefo002@ucr.edu
5   *      Lab Section: 22
6   *      Assignment: Lab #0  Exercise # Chapter 2 Try 6
7   *      Exercise Description:
8   *      Run the above program. Note that B gets assigned 203 (in binary).
9   *      I acknowledge all content contained herein, excluding template or exampl
    e
10  *      code, is my own original work.
11  */
12 #include "RIMS.h"
13 unsigned char x;
14
15 void main() {
16 while (1) {
17 x = 203;
18 B = x;
19 }
20 }
```

```
/*      adeja001_lab0_part3.c− 4/1/13
 *      Name & E−mail: Ariana Dejaco adeja001@ucr.edu
 *      CS Login: Adeja001
 *      Partner(s) Name & E−mail: Josuha DeForest−Williams jdefo002@ucr.edu
 *      Lab Section: 22
 *      Assignment: Lab #0  Exercise # Chapter 3 Try 17
 *      Exercise Description:
 *      Run the above program, change AâM−^@M−^Ys switches to different values, and observe the output LEDs. Use the step functionality to observe that only one of the three branchesâM−^@M−^Y will have its statements executed.
 *      I acknowledge all content contained herein, excluding template or example
 *      code, is my own original work.
 */#include "RIMS.h"

void main() {
while (1) {
if (A <= 100) {
B0 = 1;
B1 = 0;
B2 = 0;
}
else if (A <= 200) {
B0 = 0;
B1 = 1;
B2 = 0;
}
else {
B0 = 0;
B1 = 0;
B2 = 1;
}
}
}
```

```
1  /*      Adeja001_lab0_part4.c − 4/1/13
2   *      Name & E−mail: Ariana Dejaco adeja001@ucr.edu
3   *      CS Login: Adeja001
4   *      Partner(s) Name & E−mail: Josuha DeForest−Williams jdefo002@ucr.edu
5   *      Lab Section: 22
6   *      Assignment: Lab #0  Exercise # Chapter 3 Try 17
7   *      Exercise Description:
8   *      Run the above program. Set A to 2 and note from the printed output that
    the inner while loop body executes twice, and the result is 4, which appears on
    B. Set A to 0 and note that the inner while loop body does not execute, and the
    result is 1. Set A to 15 and note that the result is 32768, though B does not sh
    ow that value because BâM−^@M−^Ys 8 bits cannot represent numbers larger than 25
    5.
9   *      I acknowledge all content contained herein, excluding template or exampl
    e
10  *      code, is my own original work.
11  */
12 #include "RIMS.h"
13
14 void main() {
15 unsigned long result;
16 unsigned char i;
17 while (1) {
18 result = 1;
19 i = 1;
20 while (i <= A) {
21 result = result * 2;
22 puts(" Iteration: "); puti(i); puts("\r\n");
23 i = i + 1;
24 }
25 puts("Result: "); puti(result); puts("\r\n\n");
26 B = result;
27 }
28 }
```

```
1  /*      Adeja001_lab0_part5.c − 4/1/13
2   *      Name & E−mail: Ariana Dejaco adeja001@ucr.edu
3   *      CS Login: Adeja001
4   *      Partner(s) Name & E−mail: Joshua DeForest−Williams jdefo002@ucr.edu
5   *      Lab Section: 22
6   *      Assignment: Lab # 0   Exercise # Chapter 5 try # 37
7   *      Exercise Description:Run the above program and observe that the printed
    output reflects the values of A7−A0. Next, add to the above program a new functi
    on definition named AryCountOnes, having the same two parameters as AryPrint and
     with a return type of unsigned char, that returns the number of array elements
    equal to 1. Update main to also set B to that number.
8   *
9   *      I acknowledge all content contained herein, excluding template or exampl
    e
10  *      code, is my own original work.
11  */
12 #include "RIMS.h"
13
14 void ACopyToArray(unsigned char Ary[]) {
15 Ary[0]=A0; Ary[1]=A1; Ary[2]=A2; Ary[3]=A3;
16 Ary[4]=A4; Ary[5]=A5; Ary[6]=A6; Ary[7]=A7;
17 return;
18 }
19
20 void AryPrint(unsigned char Ary[], unsigned char ArySize) {
21 unsigned char i;
22 for (i=0; i<ArySize; i++) {
23 puti(Ary[(ArySize−1)−i]); // (ArySize−1)−i: print in reverse
24 putc(' ');
25 }
26 putc('\n');
27 return;
28 }
29
30 unsigned char AryCountOnes(unsigned char Ary[], unsigned char ArySize)
31 {
32     unsigned char returnVal = 0;
33     int i;
34     for( i = 0; i < ArySize; i++)
35     {
36         if(Ary[i] == 1)
37           returnVal++;
38     }
39     return returnVal;
40 }
41
42 void AryComplement(unsigned char Ary[], unsigned char ArySize)
43 {
44     int i;
45     for(i = 0; i < ArySize; i++)
46     {
47         if(Ary[i] == 0)
48           Ary[i] = 1;
49         else if(Ary[i] == 1)
50           Ary[i] = 0;
51     }
52 }
53
54 void main(){
55 const unsigned char AASize = 8;
56 unsigned char AA[8];
57 while(1) {
58 B = CountOnes(A0, A1, A2);
59 puti(count); // ADD this statement
60 }
61 }
```

```
/*      Adeja001_lab0_part6.c − 4/5/13
 *      Name & E−mail: Ariana Dejaco adeja001@ucr.edu
 *      CS Login: Adeja001
 *      Partner(s) Name & E−mail:Joshua DeForest−Williams jdefo002@ucr.edu
 *      Lab Section: 22
 *      Assignment: Lab # 0  Exercise # 1 Chapter 1
 *      Exercise Description:
 *  Write RIM C code that sets B0 to 1 only if A0−A3 are all 1's or if A5−A7 are all 1's (or if both situations are true. Using border and sample imput value combinations, test the written code with RIMS, and generate a timing diagram showing the test results
 *      I acknowledge all content contained herein, excluding template or example
 *      code, is my own original work.
 */
#include "RIMS.h"
void main(){
        while(1){
        if (A0 && A1 && A2 && A3) {
            B0 = 1;
        }
        else if (A4 && A5 && A6 && A7) {
            B0 = 1;
        }
        else {
            B0 = 0;
        }
        }
}
```

```
/*      adeja001_lab0_part7.c − 4/5/13
 *      Name & E−mail: Ariana Dejaco adeja001@ucr.edu
 *      CS Login: ADeja001
 *      Partner(s) Name & E−mail:Joshua DeForest−Williams jdefo002@ucr.edu
 *      Lab Section: 22
 *      Assignment: Lab # 0  Exercise Try # 11 Chapter 2
 *      Exercise Description:
 *  Write a single C statement for RIM that sets B3−B0 to A5−A2 and sets other o
utput bits to 0s.
 *      I acknowledge all content contained herein, excluding template or exampl
e
 *      code, is my own original work.
 */
#include "RIMS.h"

void main(){
        while(1){
        B = A >>2 ;
        }
}
```