100 points possible. Open notes, text, calculator. No laptops.

NAME: Solution

1. (5 pts each, 25 pts total) Suppose you have a CPU that is similar to the LC-3, but with a word size of 12-bits, and that the registers contain the following values:

   \[ \begin{align*}
   R0 &= \text{x983} \\
   R1 &= \text{x36B} \\
   R2 &= \text{x901} \\
   R3 &= \text{xF02}
   \end{align*} \]

   a. If the value in R3 is interpreted as a 2's complement, signed integer, what is the decimal result?

   \[ 000111111110 \]

   b. If the value in R2 is interpreted as an unsigned integer, what is the decimal result?

   \[ 9 \times 16^2 + 9 \times 16^1 + 1 \times 16^0 = 2305 \]

   c. What value would be placed in R4 by the instruction ADD R4, R0, R3 (answer in hexadecimal)

   \[ \begin{align*}
   &983 \\
   \times \text{xFO2} \\
   \hline
   &\text{x8B5}
   \end{align*} \]

   d. Did the instruction in part (c) result in a 2's complement overflow?

   Yes

   e. How many bytes of memory would the LC-3 support if addresses were 12-bits long instead of 16? Assume no other changes are made to the LC-3 architecture.

   \[ 2^6 \times 2 \text{ bytes/word} = 2^{13} \text{ bytes} \]

2. (10 pts) The following assembly program has one or more errors. Identify the error(s) and explain/show how to fix it/them.

   Code to perform \( B = A + 20 \):
   \[
   \text{LD} \quad R3, A \\
   \text{ADD} \quad R3, R3, #20 \\
   \text{ST} \quad R3, B \\
   \text{HALT}
   \]

   A .FILL xDEAD
   B .FILL xBEEF
   .END

   Could solve it by adding 15 and then 5.
3. (10 pts) Are there any values of \( f \) and \( g \) that would cause the following C code to print “False”? If your answer is yes, give an example of values for \( f \) and \( g \) that would result in the program printing “False”. If your answer is no, briefly explain why the program will never output “False”.

```c
float f = foo(); // foo() places some value in f
float g = bar(); // bar() places some value in g
if ((f + g - f) == g)
    printf("True\n");
else
    printf("False\n");
```

It can print false. If \( f \) is the largest floating point #, then the value of \( g \) will get lost.

\[ f + g - f = 0 \text{ if } f \text{ is MAX} \]

4. (10 pts each, 30 pts total) The table to the right represents a snapshot of a portion of the memory of an LC-3 computer, the values in registers R0 through R3 are given below.

<table>
<thead>
<tr>
<th>Address</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>x3000</td>
<td>xB001</td>
</tr>
<tr>
<td>x3001</td>
<td>x643A</td>
</tr>
<tr>
<td>x3002</td>
<td>x3004</td>
</tr>
<tr>
<td>x3003</td>
<td>xA1FE</td>
</tr>
<tr>
<td>x3004</td>
<td>xC3C3</td>
</tr>
</tbody>
</table>

a. If the word located at address x3000 were treated as a machine instruction and executed, what value would be placed into register R0? \( \text{Answer in hex} \)

b. If the word located at address x3001 were treated as a machine instruction and executed, what value would be placed into register R2? \( \text{Answer in hex} \)

c. If the word located at address x3003 were treated as a machine instruction and executed, what value would be placed into register R0? \( \text{Answer in hex} \)
Consider the following LC-3 assembly language program:

```
.ORIG x3000
START  LEA   R0, ARRAY
        LDR   R1, R0, #0
        ADD   R0, R0, #1
AGAIN   LDR   R2, R0, #0
        BRnz  DONE
        NOT   R3, R1
        ADD   R3, R3, #1
        ADD   R3, R3, R2
        BRnz  SKIP
ST      R1, RESULT
DONE    ST    R1, RESULT
        HALT
ARRAY  .FILL x0010
        .FILL x0003
        .FILL xF382
        .FILL x303C
        .FILL x2020
        .FILL x0000
RESULT .BLKW 1
.END
```

a. Write the machine code instruction word that will be generated for the instruction “BRnzp AGAIN” in this code (answer in hex):

```
0FF7
```

b. What hex value will be found in the memory location labeled “RESULT” after this code is executed?

```
x303C
```

c. Describe in one sentence what this program does.

```
Finds the maximum value in an array.
```

6. (10 pts) The last page of this exam shows the datapath of the LC-3 CPU. Note that input B of the ALU is supplied via the SR2MUX. One of the inputs to this MUX comes from the instruction register (IR). Which two LC-3 instructions cause data from the IR to be forwarded to the ALU?

```
ADD and AND
```
IEEE 32-bit floating point format:

```
<table>
<thead>
<tr>
<th>sign</th>
<th>exponent (8 bits)</th>
<th>fraction (23 bits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>001111100000000000000000000000000000000000000000000000000</td>
<td></td>
</tr>
</tbody>
</table>
```

=0.15625