Assignment 1 Cover Page

Structure and Application of Microcomputers

Student Information (Print)

Conor
First Name

Roberts
Last Name

1056167
Student ID

Name of Marker: ______________________ **Mark:** ______________

Instructions: This assessment is designed to emphasize and reinforce some of the important concepts related to reading, lecture, and lab materials covered in week 1. (Note: There is no official lab in week 1. Nevertheless, the lab time is used to introduce material related to fixed- and floating-point representations.) To maximize your benefit from this exercise you are encouraged to do the following:

(1) Read through the assignment questions *before* listening to the lectures.
(2) Answer as many questions as possible in tandem with viewing and listening to the lectures. You can easily pause a lecture to answer a question, or re-listen to any material that you do not fully understand.
(3) Simply writing down a solution to a particular problem is not enough to obtain full marks. You must show how you arrived at the solution by showing the step-by step procedure that you performed. This procedure can be hand-written or typed – the choice is yours. However, make sure to answer each question in the provided spaces.
(4) Review your answers a few days after completing the assignment. Taking a few minutes to do this can dramatically improve your understanding of key concepts and increase your retention.

Submission: Submission of this assessment is a two-step process that involves (1) uploading your written work as a single PDF to the Dropbox on CourseLink, and (2) entering only the final answers to the assignment questions under the Quiz Section on CourseLink. **Only assignments that complete both steps will be marked. The overall submission procedure is outline below.**

(1) Once you have filled in the cover page with your personal information and answered all of the assignment questions, scan your assignment pages (including this cover sheet and in the correct order) and convert them into a single PDF submission. If you do not have a scanner, you may use a cellphone to take pictures of each page, then use a program like Adobe Acrobat Scan to create a single PDF. (Remember to use appropriate lighting, as dim or blurry pictures that cannot be read will not be marked.) Do not upload individual pages. Upload your single PDF submission to the **Dropbox labeled Assignment 1** on CourseLink.

(2) While in CourseLink, now click on the **Quizzes tab** on the navigation bar. On the Quiz List page, you will see a quiz called **Assignment 1.** Click Start Quiz. Then, answer each question using the final answers that you have already determined. As you complete each question, the answer will be automatically saved. You can see which questions have saved answers in the Questions section of the quiz's left panel. You can also click the question number in the quiz's left panel to go back to the question. Click **Go to Submit Quiz** after you answer all quiz questions.

Assignment 1 is due **11:59pm, September 18, 2020.** As stated in the course outline, assignments can be at most 2-days late without penalty. This corresponds to 11:59pm, September 20, 2020. No additional extension will be granted for any reason.

Best of success!

When answering the following questions show your work. Simply writing down an answer is not enough to obtain full marks.

Use the following information to answer questions 1-5:

As discussed in lecture, computers do not generally interpret and assign meaning to the binary data that they store and manipulate. Rather, this task falls to the programmer.

Assume that a computer's memory contains the following sequence of eleven 8-bit (hexadecimal) values: 0x55, 0x6F, 0x47, 0xBB, 0x6E, 0x96, 0x68, 0x42, 0xC8, 0x80, 0x00.

1. Interpret the first three 8-bit values in the sequence as ASCII characters. What are the values of the characters? [1.5 marks]

55 5 → 0101
5 → 0101 = 1010101 = U

6F = 6 → 1101
F → 1111 = 1101111 = o

47 = 4 → 1000
7 → 0111 = 1000111 = G

$\boxed{U\ o\ G}$

2. Interpret the next two 8-bit values in the sequence as one unsigned 16-bit integer. Convert the integer to decimal. [1 mark]

6E $= 11\cdot16^3 + 11\cdot16^2 + 6\cdot16^1 + 14\cdot16^0 = \boxed{47982_{10}}$

(base 10)

3. Interpret the next 8-bit value as a signed, 2's-complement value. Convert the integer to decimal. [1 mark]

96 → 9 = 1001

→ 6 = 0110

= 10010110 | 01101001 (1's comp.) | 01101001 + 1 = 01101010 (2's comp.) | 01101010 → $\boxed{106_{10}}$

4. Interpret the next 8-bit value as an unsigned fixed-point fraction with an implied binary point in front of the left-most bit. Convert the fraction to decimal. [1 mark]

6 = 110

→ 8 = 1000

11010000 (binary) | 0.1101000

$2^{-1} + 2^{-2} + 1\cdot2^{-4} = \frac{13}{16} = \boxed{0.8125}$

5. Interpret the remaining four 8-bit values as one IEEE-754 single-precision floating-point value. Convert the floating-point number to decimal. [1.5 marks]

42C88000 =

0 10000101 10010001000000000000000

sign exp. mantissa

$1.10010001 \cdot 2^6$

$(-1)^0 \cdot 1.56640625 \cdot 2^6$ = 1.56640625$_{10}$

= 100.25

Use the following information to answer questions 6-8:

Interpret the left-most bit of the following three 16-bit values as a parity bit. In each case identify if the value is using "odd" or "even" parity. [1.5 marks]

6. 0x1234 = $\boxed{0001}$ 0010 0011 0100 Odd parity bit

7. 0xC432 = $\boxed{1100}$ 0100 0011 0010 Even parity bit

8. 0x8F03 = $\boxed{1000}$ 1111 0000 0011 Even parity bit

9. Add the following two binary numbers: 10101 and 1011. How many carries (of 2) are generated when performing the addition? [2 marks]

```
  10101
+  1011
-------
 100000
```

5 carries

10. Repeat the previous problem, but this time subtract the second binary number from the first. How many borrows (of 2) are required when performing the subtraction? [2 marks]

```
  10101
-  1011
-------
  01010
```

2 borrows

11. The maximum unsigned binary number that can be represented in a certain computer has a decimal value of 4095. What is the word size of this computer? [1 mark]

4095_{10} can be achieved in binary using $\boxed{12}$ bits.

$111111111111_2 = 4095_{10}$

12. A certain computer has a 12-bit word size. What is the largest positive decimal value that can be represented on this machine assuming a 2's-complement representation? [1 mark]

$2^{12-1} - 1 = 2^{11} - 1 = 2048 - 1$

$\boxed{= 2047}$

Use the following information to answer questions 13-21

A and B are integer variables in a computer program, with A = 107_{10} and B = 42_{10}. The program is running on a computer that uses 8-bit 2's-complement arithmetic.

13. Show how A is represented in binary. [1 mark]

$\boxed{01101011}$

Show how -A is represented in binary. [1 mark]

1's comp (10010100) 2's comp $\boxed{10010101}$

Show how B is represented in binary. [1 mark]

00101010

Show how -B is represented in binary. [1 mark]

1's comp (11010101) 2's comp $\boxed{11010110}$

$-A = 10010101$ $\quad A = 107_{10} = 01101011_2$

$-B = 11010110$ $\quad B = 42_{10} = 00101010_2$

14. Show how the machine would compute A+B. [1 mark]

```
  01101011
+ 00101010
----------
  10010101
```

15. Did A+B produce signed overflow? Explain. [1 mark]

Yes because the correct value (149_{10}) is out of range of -128 to 127.

16. Show how the machine would compute A-B. [1 mark]

A + (-B)

```
(A)      01101011
(-B) +   11010110
-----------------
      ~~1~~ 01000001
```

17. Did A-B produce signed overflow? Explain. [1 mark]

No because the correct value (65_{10}) is within the range of -128 to 127.

18. Show how the machine would compute -A+B. [1 mark]

```
(-A)      10010101
(B)  +    00101010
------------------
          10111111
```

19. Did -A+B produce signed overflow? Explain. [1 mark]

No because the correct value (-63_{10}) is within the range -128 to 127.

20. Show how the machine would compute -A-B. [1 mark]

```
  10010101 (-A)
+ 11010110 (-B)
---------------
~~1~~ 01101011
```

21. Did -A-B produce signed overflow? Explain. [1 mark]

Yes because the correct Value (-149) is out of range of -128 to 127.

22. A 5-bit IEEE floating-point representation has the following parameters: a single sign bit, two exponent bits (i.e., K=2) and two fraction bits (i.e., n=2). What is the exponent bias? [1 mark]

exponent bias: 1

<u>Use the following information to answer questions 23-25</u>

According to the round-to-even rule, show how the following binary fractional values would be rounded to 1 bit to the right of the binary point. In each case, show the numeric values as decimal (integer and fractional) values both before and after rounding. [1/2 mark for each before and after]

E.g., 100.010 (e.g., 4 and1/4) before; after rounding 100.0~~1~~0 (e.g., 4)

23. $100.011_2 = 4.375_{10}$ or 4 and 1/4 and 1/8
Rounded: $100.000_2 = 4_{10}$

24. $100.110_2 = 4.750_{10}$ or 4 and 1/2 and 1/4
Rounded: 101.000_2 or 5_{10}

25. $101.001_2 = 5.125_{10}$ or 5 and 1/8
Rounded: 101.000_2 or 5_{10}

26. Convert the decimal number 42.6875 to IEEE-754 Single Precision (32-bit) floating-point format. [1.5 marks]

$0.\ 1.01010\ 1011 \cdot 2^5$

exp = 127+5
$= 132_{10}$
$= 10000100_2$

$\boxed{0\ 10000100\ 01010101100000000000000}$

27. Convert the decimal number 0.09375 to IEEE-754 Single Precision (32-bit) floating-point format. [1.5 marks]

0.00011

exp = 127+0

= 127_{10}

$\boxed{0\ 01111111\ 00011000000000000000000}$ = 1111111_2

28. Add the two (32-bit) floating-point numbers from the previous two questions. (You should work with the biased exponents; don't forget the (implied) leading 1 in the mantissa.) Express the result as a new 32-bit floating-point number. [2 marks]

~~0 01111111 00011000000000000000000~~

~~0 10000100 01010101100000000000000~~

~~1 0000001 10 1 0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0~~

26) 0 10000100 010101011

132 - 127 = 5

27) 0 01111111 00011...

sign exp. mantissa

127 - 127 = 0

26) $(-1)^0 \cdot 1.010101011 \cdot 2^5$

27) $(-1)^0 \cdot 1.00011 \cdot 2^0$

$= (-1)^0 \cdot 0.0000100011 \cdot 2^5$

1.0101010110

0.0000100011

1.0101111001 $\cdot 2^5$

5+127=132

$\boxed{0\ 10000100\ 0101111001...}$