hw1-definitions-and-notation

CSE20S24

Due: 4/9/24 at 5pm (no penalty late submission until 8am next morning)

In this assignment,

You will practice reading and applying definitions to get comfortable working with mathematical language.

Relevant class material: Week 1.

You will submit this assignment via Gradescope (https://www.gradescope.com) in the assignment called “hw1-definitions-and-notation”.

For all HW assignments: These homework assignments may be done individually or in groups of up to 3 students. Please ensure your name(s) and PID(s) are clearly visible on the first page of your homework submission, start each question on a new page, and upload the PDF to Gradescope. If you’re working in a group, *submit only one submission per group*: one partner uploads the submission through their Gradescope account and then adds the other group member(s) to the Gradescope submission by selecting their name(s) in the “Add Group Members” dialog box. You will need to re-add your group member(s) every time you resubmit a new version of your assignment.

Each homework question will be graded either for **correctness** (including clear and precise explanations and justifications of all answers) or **fair effort completeness**. You may collaborate on “graded for correctness” questions only with CSE 20 students in your group; if your group has questions about a problem, you may ask in drop-in help hours or post a private post (visible only to the Instructors) on Piazza. For “graded for completeness” questions: collaboration is allowed with any CSE 20 students this quarter; if your group has questions about a problem, you may ask in drop-in help hours or post a public post on Piazza.

All submitted homework for this class must be typed. You can use a word processing editor if you like (Microsoft Word, Open Office, Notepad, Vim, Google Docs, etc.) but you might find it useful to take this opportunity to learn LaTeX. LaTeX is a markup language used widely in computer science and mathematics. The homework assignments are typed using LaTeX and you can use the source files as templates for typesetting your solutions.

Integrity reminders

- Problems should be solved together, not divided up between the partners. The homework is designed to give you practice with the main concepts and techniques of the course, while getting to know and learn from your classmates.
- You may not collaborate on homework questions graded for correctness with anyone other than your group members. You may ask questions about the homework in office hours (of the instructor, TAs, and/or tutors) and on Piazza (as private notes viewable only to the Instructors). You *cannot* use any online resources about the course content other than the class material from this quarter – this is primarily to ensure that we all use consistent notation and definitions (aligned with the textbook) and also to protect the learning experience you will have when the 'aha' moments of solving the problem authentically happen.
- Do not share written solutions or partial solutions for homework with other students in the class who are not in your group. Doing so would dilute their learning experience and detract from their success in the class.

Assigned questions

1. Modeling

(a) (*Graded for completeness*) [1] In class, we used 4-tuples to represent the user ratings for four movies. This representation is memory-efficient because we use the order of the components in the 4-tuples to represent which move is being rated. However, it is not easily extended when we want to add new movies to the database. Define a new model that would allow us to represent the user ratings of movie databases where we allow for new movies to be added. Use only the data types we have talked about in class: sets, n-tuples, and strings. Explain the design choices that you used to define your model by referencing properties of the data-type(s) you choose. Demonstrate your model by showing how the rating of the user who dislikes Dune and Oppenheimer and likes Barbie and Nimona is represented.

(b) (*Graded for completeness*) Colors can be described as amounts of red, green, and blue mixed together [2] Mathematically, a color can be represented as a 3-tuple (r, g, b) where r represents the red component, g the green component, b the blue component and where each of r, g, b must be a value from this collection of numbers:

[1] This means you will get full credit so long as your submission demonstrates honest effort to answer the question. You will not be penalized for incorrect answers. To demonstrate your honest effort in answering the question, we expect you to include your attempt to answer *each* part of the question. If you get stuck with your attempt, you can still demonstrate your effort by explaining where you got stuck and what you did to try to get unstuck.

[2] This RGB representation is common in web applications. Many online tools are available to play around with mixing these colors, e.g. `https://www.w3schools.com/colors/colors_rgb.asp`.

$$\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255\}$$

(This is the same definition as in the Week 1 Review quiz.)

Can you find two different 3-tuples that represent colors that are indistinguishable to your eye? You can use the website in the footnote to play around with different choices of red, green, and blue levels to see if you can distinguish between the resulting colors. Why or why not?

A complete answer will include the specific example 3-tuples that work, along with a description of the colors that they represent and why they are indistinguishable, or an explanation of why there can't be such an example.

2. Sets and functions

   (a) (*Graded for correctness*) [3] Each of the sets below is described using set builder notation or recursion or as a result of set operations applied to other known sets. Rewrite each of the sets using the roster method.

   Remember our discussions of data-types: use clear notation that is consistent with our class notes and definitions to communicate the data-types of the elements in each set.

   Sample response that can be used as reference for the detail expected in your answer:

   The set $\{\texttt{A}\} \circ \{\texttt{AU}, \texttt{AC}, \texttt{AG}\}$ can be written using the roster method as

   $$\{\texttt{AAU}, \texttt{AAC}, \texttt{AAG}\}$$

   because set-wise concatenation gives a set whose elements are all possible results of concatenating an element of the left set with an element of the right set. Since the left set in this example only has one element, namely $\texttt{A}$, each of the elements of the set we described starts with $\texttt{A}$. There are three elements of this set, one for each of the distinct elements of the right set.

[3] This means your solution will be evaluated not only on the correctness of your answers, but on your ability to present your ideas clearly and logically. You should explain how you arrived at your conclusions, using mathematically sound reasoning. Whether you use formal proof techniques or write a more informal argument for why something is true, your answers should always be well-supported. Your goal should be to convince the reader that your results and methods are sound.

i.
$$\{n \in \mathbb{Z}^+ \mid n \leq 3\} \times \{m \in \mathbb{N} \mid m \leq 3\}$$

(Note: typo fixed Apr 3)

ii. The set X defined recursively as

Basis Step: $1 \in X, 3 \in X, 5 \in X$
Recursive Step: If the integer $n \in X$, then the result of multiplying n by -1 is in X

iii.
$$\{x \in S \mid rnalen(x) = 2\} \circ \{x \in S \mid rnalen(x) = 0\}$$

where S is the set of RNA strands and $rnalen$ is the recursively defined function that we discussed in class,

		$rnalen : S$	$\to \mathbb{Z}^+$
Basis Step:	If $b \in B$ then	$rnalen(b)$	$= 1$
Recursive Step:	If $s \in S$ and $b \in B$, then	$rnalen(sb)$	$= 1 + rnalen(s)$

iv.
$$\{(r, g, b) \in C \mid r + g + b = 2 \text{ and } g = 1\}$$

where $C = \{(r, g, b) \mid 0 \leq r \leq 255, 0 \leq g \leq 255, 0 \leq b \leq 255, r \in \mathbb{N}, g \in \mathbb{N}, b \in \mathbb{N}\}$ is the set that you worked with in Monday's review quiz.

(b) (*Graded for correctness*) Recall the function which takes an ordered pair of ratings 4-tuples and returns a measure of the difference between them

$$d_0 : \{-1, 0, 1\}^4 \times \{-1, 0, 1\}^4 \to \mathbb{R}$$

given by

$$d_0(\ (\ (x_1, x_2, x_3, x_4), (y_1, y_2, y_3, y_4)\ )\ ) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2 + (x_4 - y_4)^2}$$

Define a **new** function which we'll call d_{new} with the same domain $\{-1, 0, 1\}^4 \times \{-1, 0, 1\}^4$ and codomain $\mathbb{R}$ but where there is some example pair of ratings 4-tuples

$$(\ (x_1, x_2, x_3, x_4), (y_1, y_2, y_3, y_4)\ )$$

where

$$d_0(\ (\ (x_1, x_2, x_3, x_4), (y_1, y_2, y_3, y_4)\ )\ ) \neq d_{new}(\ (\ (x_1, x_2, x_3, x_4), (y_1, y_2, y_3, y_4)\ )\ )$$

Your answer should include **both** a precise and clear definition for the rule defining d_{new} which unambiguously specifies output for each input of the function **and** the example ordered pair of ratings 4-tuples that demonstrate that the functions are not equal. Also include a justification of your answers with (clear, correct, complete) calculations for each of the function applications and/or references to definitions and connecting them with the desired conclusion.

(c) (*Graded for correctness*) A function *basecount* that computes the number of a given base b appearing in a RNA strand s is defined recursively:

$$\begin{array}{lll}
 & basecount : S \times B & \to \mathbb{N} \\
\text{Basis Step:} & & \\
\text{If } b_1 \in B, b_2 \in B & basecount(\ (b_1, b_2)\ ) & = \begin{cases} 1 & \text{when } b_1 = b_2 \\ 0 & \text{when } b_1 \neq b_2 \end{cases} \\
\text{Recursive Step:} & & \\
\text{If } s \in S, b_1 \in B, b_2 \in B & basecount(\ (sb_1, b_2)\ ) & = \begin{cases} 1 + basecount(\ (s, b_2)\ ) & \text{when } b_1 = b_2 \\ basecount(\ (s, b_2)\ ) & \text{when } b_1 \neq b_2 \end{cases}
\end{array}$$

Consider the function application

$$basecount(\ (\texttt{ACAU}, \texttt{A})\ )$$

What is the input? What is the output? Give an example of a different choice of input that gives the same output.

Your answer should include clearly labeled answers to each of the three parts of the question, along with a justification for the values of the applications that makes specific reference to the parts of the recursive definition of the *basecount* function used to calculate it.

hw2-numbers Due: 4/16/24 at 5pm (no penalty late submission until 8am next morning)

In this assignment,

You will practice applying functions and tracing algorithms in multiple contexts, and exploring properties of positional number representations.

Relevant class material: Week 2.

You will submit this assignment via Gradescope (https://www.gradescope.com) in the assignment called "hw2-numbers".

For all HW assignments: These homework assignments may be done individually or in groups of up to 3 students. Please ensure your name(s) and PID(s) are clearly visible on the first page of your homework submission, start each question on a new page, and upload the PDF to Gradescope. If you're working in a group, *submit only one submission per group*: one partner uploads the submission through their Gradescope account and then adds the other group member(s) to the Gradescope submission by selecting their name(s) in the "Add Group Members" dialog box. You will need to re-add your group member(s) every time you resubmit a new version of your assignment.

Each homework question will be graded either for **correctness** (including clear and precise explanations and justifications of all answers) or **fair effort completeness**. You may collaborate on "graded for correctness" questions only with CSE 20 students in your group; if your group has questions about a problem, you may ask in drop-in help hours or post a private post (visible only to the Instructors) on Piazza. For "graded for completeness" questions: collaboration is allowed with any CSE 20 students this quarter; if your group has questions about a problem, you may ask in drop-in help hours or post a public post on Piazza.

All submitted homework for this class must be typed. You can use a word processing editor if you like (Microsoft Word, Open Office, Notepad, Vim, Google Docs, etc.) but you might find it useful to take this opportunity to learn LaTeX. LaTeX is a markup language used widely in computer science and mathematics. The homework assignments are typed using LaTeX and you can use the source files as templates for typesetting your solutions.

Integrity reminders

- Problems should be solved together, not divided up between the partners. The homework is designed to give you practice with the main concepts and techniques of the course, while getting to know and learn from your classmates.
- You may not collaborate on homework questions graded for correctness with anyone other than your group members. You may ask questions about the homework in office hours (of the instructor, TAs, and/or tutors) and on Piazza (as private notes viewable only to the Instructors). You *cannot* use any online resources about the course content other than the

class material from this quarter – this is primarily to ensure that we all use consistent notation and definitions (aligned with the textbook) and also to protect the learning experience you will have when the 'aha' moments of solving the problem authentically happen.

- Do not share written solutions or partial solutions for homework with other students in the class who are not in your group. Doing so would dilute their learning experience and detract from their success in the class.

Assigned questions

1. Functions and algorithms: in this question you'll explore the properties of the (integer) power and logarithm functions. We'll use some definitions we introduced in class this week, namely the function b^i with domain $\mathbb{Z}^+ \times \mathbb{N}$ and codomain $\mathbb{N}$ defined recursively by

   Basis Step:
   $b^0 = 1$
   Recursive Step:
   If $i \in \mathbb{N}, b^{i+1} = b \cdot b^i$

   and the algorithm:

   Calculating integer part of base b logarithm

```
procedure logb($b$, $n$: positive integers with $b > 1$)
$i$ := 0
while $n > b - 1$
  $i$ := $i + 1$
  $n$ := $n$ div $b$
return $i$ {$i$ holds the integer part of the base $b$ logarithm of $n$}
```

   (a) (*Graded for correctness*) [4] Choose a positive integer b between 3 and 6 (inclusive) and choose a nonnegative integer i between 2 and 5 (inclusive). Demonstrate how to calculate the result of the *logb* algorithm (procedure) when its input is the base b you chose and $n = b^i$ (for the i you choose.) A complete answer will include the specific choice of b and i, along with trace of the calculations of b^i and the computation of the algorithm, including (clear, correct, complete) calculations for each of the function applications and/or references to definitions and a trace table for algorithm that includes the values of all relevant variables at each iteration (See the annotated Week 2 notes for the level of detail expected in a trace of a function application and a trace table).

[4] This means your solution will be evaluated not only on the correctness of your answers, but on your ability to present your ideas clearly and logically. You should explain how you arrived at your conclusions, using mathematically sound reasoning. Whether you use formal proof techniques or write a more informal argument for why something is true, your answers should always be well-supported. Your goal should be to convince the reader that your results and methods are sound.

(b) (*Graded for correctness*) Now we'll go in other order: Demonstrate how to calculate the result of 7^y where y is the result of the *logb* algorithm (procedure) when its input is $b = 7$ and $n = 30$. A complete answer will include clearly labelled traces of the calculations of b^i and the computation of the algorithm, including (clear, correct, complete) calculations for each of the function applications and/or references to definitions and a trace table for algorithm that includes the values of all relevant variables at each iteration (See the annotated Week 2 notes for the level of detail expected in a trace of a function application and a trace table).

(c) (*Graded for completeness*) [5] Logarithms and powers are supposed to "undo" one another. Explain whether your work in parts (a) and (b) supports that idea, whether you saw anything confusing or surprising about these calculations, and how to explain what you saw.

2. Base expansions

(a) (*Graded for completeness*) Pick an integer between 50 and 1000 (inclusive) that you (or one of your group members) came across at some point this week. In a sentence or two, give some context for why you're choosing this number). Write the base expansion of your chosen number in base 2 (binary), base 3 (ternary), base 4, and base 16 (hexadecimal).

(b) (*Graded for correctness*) What is the **smallest** width w in which they could write your chosen number in base 8 (octal) fixed-width w? Justify your answer with reference to the definitions of fixed-width expansions and relevant calculations.

(c) (*Graded for correctness*) Express in roster method the set of numbers between 1 and 2 (exclusive) that be written without error (full precision) in binary fixed width expansion with integer part width 3 and fractional part width 2. Justify your answer with reference to the definitions of fixed-width expansions and relevant calculations.

(d) (*Graded for correctness*) Consider the strings of 1s that have length 3, 5, 7, and 10. Calculate the numbers

$[111]_{s,3}$
$[11111]_{s,5}$
$[1111111]_{s,7}$
$[1111111111]_{s,10}$
$[111]_{2c,3}$
$[11111]_{2c,5}$
$[1111111]_{2c,7}$
$[1111111111]_{2c,10}$

Justify your answers with specific reference to the definitions of sign-magnitude and 2s complement expansions and relevant calculations.

[5] This means you will get full credit so long as your submission demonstrates honest effort to answer the question. You will not be penalized for incorrect answers. To demonstrate your honest effort in answering the question, we expect you to include your attempt to answer *each* part of the question. If you get stuck with your attempt, you can still demonstrate your effort by explaining where you got stuck and what you did to try to get unstuck.

(e) (*Graded for completeness*) What patterns do you notice in your calculations in part (d)?

3. Multiple representations

Recall that, mathematically, a color can be represented as a 3-tuple (r, g, b) where r represents the red component, g the green component, b the blue component and where each of r, g, b must be from the collection $\{x \in \mathbb{N} \mid 0 \leq x \leq 255\}$. As an alternative representation, in this assignment we'll use base b fixed-width expansions to represent colors as individual numbers (this definition was introduced in this week's Review quiz).

Definition: A **hex color** is a nonnegative integer, n, that has a base 16 fixed-width 6 expansion

$$n = (r_1r_2g_1g_2b_1b_2)_{16,6}$$

where $(r_1r_2)_{16,2}$ is the red component, $(g_1g_2)_{16,2}$ is the green component, and $(b_1b_2)_{16,2}$ is the blue.

For this question, let's call the set of hex colors H. In the Week 2 Review quiz (Question 3d), we explored a few different set builder definitions for H.

(a) (*Graded for completeness*) Rewrite the set builder definition of a set below so that it refers to colors rather than numbers: $\{c \in H \mid c \mod 256 = 0\}$. A complete answer will justify the new set builder definition by connection with the definition of hex colors and how it impacts the colors that satisfy the specific property for this set.

(b) (*Graded for correctness*) Rewrite the set builder definition of a set below so that it refers to colors rather than numbers: $\{c \in H \mid c < 65536\}$. A complete answer will justify the new set builder definition by connection with the definition of hex colors and how it impacts the colors that satisfy the specific property for this set.

(c) (*Graded for correctness*) In art, we can mix two colors to get a new color. For example, red and blue make purple, blue and yellow make green, red and yellow make orange. A mathematical definition of **hex color mixing** would be a function with domain $H \times H$ and codomain H where the result of applying this function to an input (n_1, n_2) is the hex color that results from mixing the hex colors n_1 and n_2.

Sample work for a related question that can be used as reference for the detail expected in your answer: Consider the attempted mathematical definition $f_1 : H \times H \to H$ given by

$$f_1(\ (n_1, n_2)\ ) = n_1 + n_2$$

We will show that this function is not well-defined so it cannot give a hex color mixing definition. The reason it is not well-defined is that the application of the rule sometimes gives values that are not in the stated codomain. To see this, we need an example input to f_1 for which the output of the rule is not in H. Here is one such example: $(n_1, n_2) = (\ (FFFFFF)_{16,6}, (FFFFFF)_{16,6}\ )$. This example is in $H \times H$ because $(FFFFFF)_{16,6}$ is a nonnegative integer that has a base 16 fixed-width 6 expansion so it is a hex color. We now apply the definition of the rule in f_1:

$$\begin{aligned} f_1(\ (n_1,n_2)\ ) &= f_1((\ (FFFFFF)_{16,6},(FFFFFF)_{16,6}\ )) = (FFFFFF)_{16,6} + (FFFFFF)_{16,6} \\ &= 2\left(15\cdot 16^5 + 15\cdot 16^4 + 15\cdot 16^3 + 15\cdot 16^2 + 15\cdot 16^1 + 15\cdot 16^0\right) \\ &= 2\cdot 15\cdot(1048576+65536+4096+256+16+1) = 2\cdot 15\cdot 1118481 = 33554430 \end{aligned}$$

This is not a hex color because it is greater than or equal to $16^6 = 16777216$, so (using the Week 2 notes on which numbers can be represented with fixed-width expansions) it does not have a hexadecimal **fixed width** 6 expansion.

In this question, we'll look at another attempted mathematical definition for hex color mixing. We define $f_2 : H \times H \to H$ given by

$$f_2(\ (n_1,n_2)\ ) = (n_1+n_2)\ \mathbf{div}\ 2$$

This is a well-defined function. You do not need to prove this or hand it in, but it's good practice to make sure you understand why it's well-defined. Notice that the function application

$$f_2(\ (\ (FF0000)_{16,6},(FFFE00)_{16,6}\ )\ )$$

can be calculated as:

$$\begin{aligned} &f_2(((FF0000)_{16,6},(FFFE00)_{16,6})) = ((FF0000)_{16,6} + (FFFE00)_{16,6}))\ \mathbf{div}\ 2 \\ &\quad = \Big(\ \left(15\cdot 16^5 + 15\cdot 16^4 + 0\cdot 16^3 + 0\cdot 16^2 + 0\cdot 16^1 + 0\cdot 16^0\right) \\ &\qquad +\ \left((15\cdot 16^5 + 15\cdot 16^4 + 15\cdot 16^3 + 14\cdot 16^2 + 0\cdot 16^1 + 0\cdot 16^0\right)\ \Big)\ \mathbf{div}\ 2 \\ &\quad = (16711680 + 16776704)\ \mathbf{div}\ 2 = 33488384\ \mathbf{div}\ 2 = 16744192 \end{aligned}$$

Since this is less than $16^6 = 16777216$, we can represent it using base 16 fixed-width 6:.

$$(16744192)_{10} = 15\cdot 16^5 + 15\cdot 16^4 + 7\cdot 16^3 + 15\cdot 16^2 + 0\cdot 16^1 + 0\cdot 16^0 = (FF7F00)_{16,6}$$

Using the web tool `https://www.w3schools.com/colors/colors_rgb.asp`, we can verify that $(FF0000)_{16,6}$ is red, $(FFFE00)_{16,6}$ is yellow, and $(FF7F00)_{16,6}$ is orange.

Show that f_2 does not work as a hex color mixing definition by finding another ordered pair of hex colors for which the result of applying f_2 does not give the expected hex color. Include the numerical description of each colors you mention, alongside a description of them in English. Describe how you know what these colors are (if you use a web color tool, include its URL in your submission writeup; if not, describe your reasoning). Justify your example with (clear, correct, complete) calculations and/or references to definitions, and connecting them with the desired conclusion.

hw3-circuits-and-logic Due: 4/23/24 at 5pm (no penalty late submission until 8am next morning)

In this assignment, you will consider how circuits and logic can be used to represent mathematical claims. You will use propositional operators to express and evaluate these claims.

Relevant class material: Week 2 and Week 3.

You will submit this assignment via Gradescope (https://www.gradescope.com) in the assignment called "hw3-circuits-and-logic".

For all HW assignments: These homework assignments may be done individually or in groups of up to 3 students. Please ensure your name(s) and PID(s) are clearly visible on the first page of your homework submission, start each question on a new page, and upload the PDF to Gradescope. If you're working in a group, *submit only one submission per group*: one partner uploads the submission through their Gradescope account and then adds the other group member(s) to the Gradescope submission by selecting their name(s) in the "Add Group Members" dialog box. You will need to re-add your group member(s) every time you resubmit a new version of your assignment.

Each homework question will be graded either for **correctness** (including clear and precise explanations and justifications of all answers) or **fair effort completeness**. You may collaborate on "graded for correctness" questions only with CSE 20 students in your group; if your group has questions about a problem, you may ask in drop-in help hours or post a private post (visible only to the Instructors) on Piazza. For "graded for completeness" questions: collaboration is allowed with any CSE 20 students this quarter; if your group has questions about a problem, you may ask in drop-in help hours or post a public post on Piazza.

All submitted homework for this class must be typed. You can use a word processing editor if you like (Microsoft Word, Open Office, Notepad, Vim, Google Docs, etc.) but you might find it useful to take this opportunity to learn LaTeX. LaTeX is a markup language used widely in computer science and mathematics. The homework assignments are typed using LaTeX and you can use the source files as templates for typesetting your solutions.

Integrity reminders

- Problems should be solved together, not divided up between the partners. The homework is designed to give you practice with the main concepts and techniques of the course, while getting to know and learn from your classmates.
- You may not collaborate on homework questions graded for correctness with anyone other than your group members. You may ask questions about the homework in office hours (of the instructor, TAs, and/or tutors) and on Piazza (as private notes viewable only to the Instructors). You *cannot* use any online resources about the course content other than the class material from this quarter – this is primarily to ensure that we all use consistent notation and definitions (aligned with the textbook) and also to protect the learning experience

you will have when the 'aha' moments of solving the problem authentically happen.

- Do not share written solutions or partial solutions for homework with other students in the class who are not in your group. Doing so would dilute their learning experience and detract from their success in the class.

Assigned questions

1. Fixed-width addition.

   (a) (*Graded for completeness*) [6] Choose example width 5 first summand and width 5 second summand so that in the binary fixed-width addition (adding one bit at time, using the usual column-by-column and carry arithmetic, and ignoring the carry from the leftmost column), the example satisfies all three conditions below simultaneously

   (1) When interpreting each of the summands and the result in binary fixed-width 5, the result represents the actual value of the sum of the summands **and**

   (2) when interpreting each of the summands and the sum in sign-magnitude width 5, the result represents the actual value of the sum of the summands **and**

   (3) when interpreting each of the summands and the sum in 2s complement width 5, the result represents the actual value of the sum of the summands.

   (b) (*Graded for correctness*) [7] Choose an example width 5 first summand and second summand so that in the binary fixed-width addition (adding one bit at time, using the usual column-by-column and carry arithmetic, and ignoring the carry from the leftmost column), the example satisfies all three conditions below simultaneously

   (1) When interpreting each of the summands and the result in binary fixed-width 5, the result **does not** represent the actual value of the sum of the summands **and**

   (2) when interpreting each of the summands and the sum in sign-magnitude width 5, the result **does not** represents the actual value of the sum of the summands **and**

   (3) when interpreting each of the summands and the sum in 2s complement width 5, the result represents the actual value of the sum of the summands.

   A complete solution will clearly specify each summand and the result of binary fixed-width addition with this choice of summands; will specify the value of each summand and the result

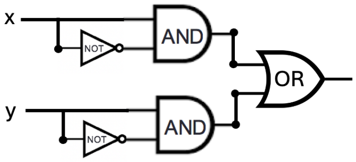
[6] This means you will get full credit so long as your submission demonstrates honest effort to answer the question. You will not be penalized for incorrect answers. To demonstrate your honest effort in answering the question, we expect you to include your attempt to answer *each* part of the question. If you get stuck with your attempt, you can still demonstrate your effort by explaining where you got stuck and what you did to try to get unstuck.

[7] This means your solution will be evaluated not only on the correctness of your answers, but on your ability to present your ideas clearly and logically. You should explain how you arrived at your conclusions, using mathematically sound reasoning. Whether you use formal proof techniques or write a more informal argument for why something is true, your answers should always be well-supported. Your goal should be to convince the reader that your results and methods are sound.

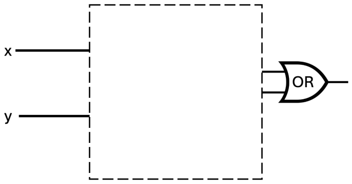
for binary fixed-width 5, sign-magnitude width 5, and 2s complement width 5 (and include calculations connecting with the definitions of these representations to explain these values); and a conclusion connecting the calculations to the properties laid out in the question.

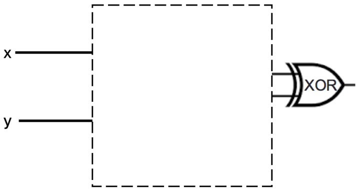
2. Circuits.

   (a) (*Graded for completeness*) Consider the circuit below with inputs x and y. Identify a pair of gates that could be switched without changing the input-output table of the circuit. If you do, write out the input-output table that results, and briefly explain why this choice of gates works. If there is no such pair of gates, explain why not with reference to the definitions of the logic gates.



   (b) (*Graded for correctness*) Is there a way to fill in the blank portion of the two logic circuits below *with the same gates connected in the same way* so that the resulting circuits have the same input-output value *even though* one uses an OR gate at the end and the other uses an XOR gate? If so, design the circuit that would be used, write out the input-output table that results, and briefly explain why your design works. If not, explain why not with reference to the definitions of the logic gates.





3. Compound propositions. The set of strings of length 4 whose characters are 0s or 1s is the result of four successive set-wise concatenations: $\{0,1\} \circ \{0,1\} \circ \{0,1\}$. Let's call this set X_4. Consider the function $f : X_4 \to X_4$ defined by

$$f(x) = \begin{cases} y & \text{when } (x)_{2,4} < 15 \text{ and } (y)_{2,4} = (x)_{2,4} + 1 \\ 1111 & \text{when } x = 1111 \end{cases}$$

for each $x \in X_4$. In other words, we can describe the function as: f takes a string, interprets it as the binary fixed-width 4 expansion of an integer, and then adds 1 to that integer (unless x is already representing the greatest integer that can be represented in binary fixed-width 4) and outputs the binary fixed-width 4 expansion of the result.

(a) (*Graded for completeness*) Fill in the blanks in the following input-output definition table with four inputs x_3, x_2, x_1, x_0 and four outputs y_3, y_2, y_1, y_0 so that $f(x_3x_2x_1x_0) = y_3y_2y_1y_0$.

x_3	x_2	x_1	x_0	y_3	y_2	y_1	y_0
1	1	1	1	1	1	BLANK1	1
1	1	1	0	1	1	1	1
1	1	0	1	1	BLANK2	1	0
1	1	0	0	1	1	0	1
1	0	1	1	1	1	0	0
1	0	1	0	1	0	1	1
1	0	0	1	1	0	1	0
1	0	0	0	BLANK3	0	0	1
0	1	1	1	1	0	0	0
0	1	1	0	0	1	1	BLANK4
0	1	0	1	0	1	1	BLANK5
0	1	0	0	0	1	0	1
0	0	1	1	0	1	0	0
0	0	1	0	0	0	1	1
0	0	0	1	0	0	1	0
0	0	0	0	BLANK6	0	0	1

(b) (*Graded for correctness*) Construct an expression (as a compound proposition) for y_0 in terms of the inputs x_3, x_2, x_1, x_0. Justify your expression by referring to the definition of the logic gates XOR, AND, OR, NOT and the definition of the function f. Hint: our work on the half-adder might be helpful.

(c) (*Graded for correctness*) Construct an expression (as a compound proposition) for y_1 in terms of the inputs x_3, x_2, x_1, x_0. Justify your expression by referring to the definition of the logic gates XOR, AND, OR, NOT and the definition of the function f. Hint: our work on the half-adder might be helpful.

(d) (*Graded for completeness*) Draw a combinatorial circuit corresponding to these compound propositions. Remember that the symbols for the inputs will be on the left-hand-side and the symbol for the outputs y_0 and y_1 will be on the right-hand side. Use gates (draw the appropriate shapes and add labels for clarity) and wires to connect the inputs appropriately to give the output.

(e) (*Graded for completeness*) Construct expressions (as a compound propositions) for y_2 and y_3 in terms of the inputs x_3, x_2, x_1, x_0. Are these similar to the expressions for y_0 and y_1?

4. Logical Equivalence. Imagine a friend suggests the following argument to you: “The compound proposition

$$(x \vee y) \wedge z$$

is logically equivalent to

$$x \vee (y \wedge z)$$

because I can transform one to the other using the following sequence of logical equivalences:

$$(x \vee y) \wedge z \equiv (x \vee (y \wedge y)) \wedge z \equiv x \vee ((y \vee y) \wedge z) \equiv x \vee (y \wedge z)$$

because y is logically equivalent to both $y \wedge y$ and to $y \vee y$”.

(a) (*Graded for correctness*) Prove to your friend that they made a mistake by giving a truth assignment to the propositional variables x, y, z so that the two compound propositions $(x \vee y) \wedge z$ and $x \vee (y \wedge z)$ have different truth values. Justify your choice by evaluating these compound propositions using the definitions of the logical connectives and include enough intermediate steps so that a student in CSE 20 who may be struggling with the material can still follow along with your reasoning.

(b) (*Graded for completeness*) Help your friend find the problem in their argument by pointing out which step(s) were incorrect.

(c) (*Graded for completeness*) Give **three** different compound propositions that are actually logically equivalent to (and not the same as)

$$(x \vee y) \wedge z$$

Justify each one of these logical equivalences either by applying a sequence of logical equivalences or using a truth table. Notice that you can use other logical operators (e.g. $\neg, \vee, \wedge, \oplus, \rightarrow, \leftrightarrow$) when constructing your compound propositions.

Bonus; not for credit (do not hand in): How would you translate each of the equivalent compound propositions in English? Does doing so help illustrate why they are equivalent?