Reading. Read the following sections in the textbook.

- Ch 1: Intro, 1.1–1.7

Part 1. Convert each mnemonic-based LMC program below into a numeric LMC program.

(1). Program:

```
INP
STA x
INP
STA y
SUB x
BRP reverse
LDA x
OUT
LDA y
OUT
BRA done

reverse LDA y
OUT
LDA x
OUT
done HLT
x DAT 000
y DAT 000
```

(2). Program:

```
INP
STA op1
INP
STA op2
loop LDA op2
BRZ done
SUB one
STA op2
LDA SUM
ADD op1
STA sum
```
(3). Program:

```
INP
BRZ done
STA n
INP
STA r
loop
INP
ADD r
OUT
LDA i
ADD one
STA i
LDA n
SUB i
BRP loop
done
HLT
one
DAT 001
i
DAT 001
n
DAT 000
r
DAT 000
```

Part 2. For each LMC program in part 1, describe in plain English what the program does and give C++ code that performs the same computation. Note that it will likely be helpful to trace the programs above to determine what they are doing.

(4). Describe and give C++ code for program (1).

(5). Describe and give C++ code for program (2).

(6). Describe and give C++ code for program (3).

Part 3. Write your own LMC programs based on the specifications below. Give the mnemonic version of each program (with labels as appropriate). For each program, provide at least one trace
to show that your program works correctly. Note that there are a number of LMC simulators online (e.g., https://www.101computing.net/LMC). While many of these support more advanced features (such as negative numbers and numbers larger than three digits), they may be helpful for designing and testing your programs below.

(7). Modify program (2) in Part 1 so that the smaller value input by the user is stored in op2.

(8). Write an LMC program that reads in \( n \) numbers from the user and then outputs the sum of the numbers. The first number input should be \( n \) (the number of values to be summed), followed by each of the \( n \) values to sum.

(9). Write an LMC program to perform integer division of two numbers input by the user (where the first number input should be divided by the second number input). For example, using integer division, \( 10/4 \) is 2, \( 10/3 \) is 3, and \( 10/6 \) is 1. Your program should output the result of the division, except in the case when the second number is zero (i.e., when the division is undefined).

(10). Write an LMC program to check if a given list of numbers is in sorted order (from smallest to largest value). If the list is sorted, your program should output 1, otherwise your program should output 0. Assume the length of the list of numbers is given at a mailbox labeled with \( n \), the mailbox containing the mailbox address of the first number in the list is labeled with \( p \), and the list of numbers are in sequential mailboxes. For instance, the list of numbers 7, 8, 9 might be represented as:

```
... 
n DAT 003
p DAT 035
  ... 
  DAT 007 # at mailbox address 35
  DAT 008 # at mailbox address 36
  DAT 009 # at mailbox address 37
```

**What to turn in.** Please submit your answers to the above questions in class on the due date. Note that you can either provide typed or handwritten answers. However, your answers must be legible and clearly marked. Be sure to include your name and the date on your work.