1. Are the following Boolean algebra expressions true or false? If true, provide a proof. If false, provide a counter example.

   (a) $a' \vee b' = (a \vee b)'$
   (b) $a \vee (a \wedge b) = a$
   (c) $(a \wedge c) \vee (a \wedge b \wedge c) = a \wedge c$

(5+5+5)

(a) False: Take $a=0$, $b=1$. Then

$$a' \vee b' = 1 \vee 0 = 1$$

$$(a \vee b)' = (0 \vee 1)' = 0$$

(b) True: $a \vee (a \wedge b) = (a \wedge 1) \vee (a \wedge b)$ (identity)

$= a \wedge (1 \vee b)$ (distributivity)

$= a \wedge 1 = a$

(c) True: $(a \wedge c) \vee (a \wedge b \wedge c) = (a \wedge c) \vee ((a \wedge c) \wedge b)$ (associativity)

$= a \wedge c$ (by part (b))

2. bfloat16 is a 16-bit floating point format where the bits are allocated as follows.

sign	exponent (8 bits)							significant (7 bits)						
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Per IEEE 754 convention, the exponent bias is 127.

(a) Convert the data word

$$1\,01111111\,0100000$$

from bfloat16 to decimal.

(b) What is the maximal relative error when converting an arbitrary decimal to bfloat16?

(You may state your answer as a fraction or in terms of a power of 2; no need to convert to decimal.)

(c) What is special about bfloat16 and why is it getting popular in neural network architectures?

(5+5+5)

(a) sign: 1 $\Rightarrow$ number is negative

exponent: $exp = (0111\ 1111)_2 - \text{bias} = 127 - 127 = 0$

significant: number is normal, so significant $= (1.01)_2 = 2^0 + 2^{-2} = 1.25$

$\Rightarrow$ number is $-1.25 \cdot 2^0 = -1.25$

(b) Rel. rounding error $\leq \frac{1}{2}$ ulp $= \frac{1}{2} \cdot 2^{-7} = 2^{-8} = \frac{1}{256} \approx 0.5\%$

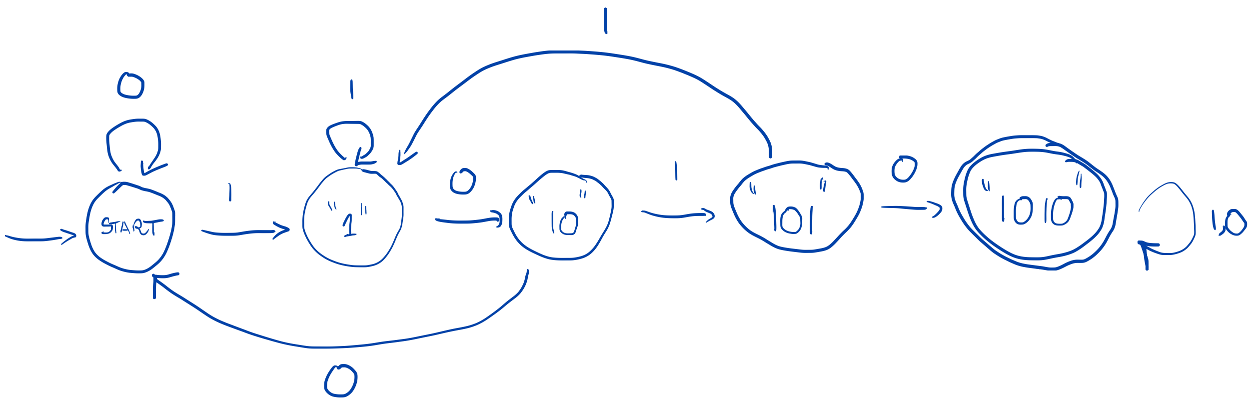
(c) • Only 2 bytes per number $\Rightarrow$ minimizes memory requirements

• Big dynamic range: $\approx 2^{-127} \ldots 2^{128}$

• But not much precision

For neural networks, lack of precision can often be compensated as training optimizes over a large number of such weights.

3. Draw a deterministic (!) finite state machine that accepts input strings that contain the substring 1010. (10)



4. (a) In a RAID-5 array, one disk out of 5 has failed. The others contain the bit sequences 11111111…, 01101000…, 00000000…, 11100001…. Reconstruct the beginning of the bit sequence of the failed disk.

(b) The RAID-5 array consists of disks with a capacity of 1 TB each and a rate of *unrecoverable read errors* (URE), defined as the the probability of a single bit error during read, of URE $= 10^{-14}$. Assuming that single-bit errors are statistically independent, estimate the probability that the array can be fully recovered.

(Simplify your answer as much as reasonably possible, but you do not need to evaluate to a number; take $1\,\text{TB} = 2^{40}\,\text{B}$.)

(c) Assume that the hard drives have a native block size of $4\,\text{kB} = 2^{12}\,\text{B}$. A block can only be read fully, or not at all. What is the rate of unrecoverable block reads? Use this number to explain why adding another disk and running the array in RAID-6 mode makes the recovery process much safer.

(5+5+5)

(a) 01110110 …

(b) Need to read $4 \cdot 2^{40} \cdot 8 = 2^{45}$ bits

Thus, we expect $10^{-14} \cdot 2^{45}$ bit errors on average.

If this number is $\ll 1$, it is approximately the probability of an error during recovery. So $P(\text{full recovery}) \approx 1 - 10^{-14} \cdot 2^{45}$ (≈ 0.65).

(c) The probability of multiple errors during block read is negligible, so the rate of unrecoverable block errors is $10^{-14} \cdot 2^{12} \cdot 8 = 10^{-14} \cdot 2^{15}$

Now suppose that RAID-6 recovery hits an unrecoverable read error. Then recovery only fails if the corresponding block on any of the remaing disks fails, with a probability of

$$4 \cdot 10^{-14} \cdot 2^{15} = 10^{-14} \cdot 2^{17} \quad (\approx 1.3 \cdot 10^{-9})$$

which is very low. (Actual risk will be larger as errors may be correlated…)

5. (a) Encode the data word

$$1001\,0111\,010$$

in Hamming-(16,11) encoding. Use the bit-ordering convention employed in class.

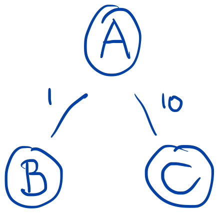
(5)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
p_0	p_1	p_2	d_0	p_4	d_1	d_2	d_3	p_8	d_4	d_5	d_6	d_7	d_8	d_9	d_{10}
$\boxed{1}$	$\boxed{1}$	$\boxed{1}$	1	$\boxed{1}$	0	0	1	$\boxed{0}$	0	1	1	1	0	1	0

6. Give a simple example of a network with distance vector routing that illustrates the saying "good news travels fast, bad news travels slow".

Your answer should provide details of a "good news" and a "bad news" event. (10)

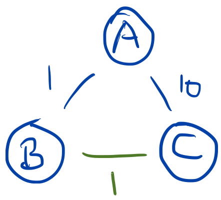
(i)



Costs:

	A	B	C
A	0	1	10
B	1	0	11
C	10	11	0

(ii) Now B discovers a link at cost 1 to C:



Now B's distance vector is $\begin{array}{ccc} A & B & C \\ (1 & 0 & 1) \end{array}$

and it takes one broadcast for A and C to update:

	A	B	C
A	0	1	2
B	1	0	1
C	2	1	0

(iii) Now suppose A–B goes down. Without route poisoning:

- B computes $d_B(A) = d_B(C) + d_C(A) = 1+2=3$ and broadcasts
- C computes $d_C(A) = d_C(B) + d_B(A) = 1+3=4$ and broadcasts
- B computes $d_B(A) = d_B(C) + d_C(A) = 1+4=5$ ......

This continues until the direct link C–A is cheaper, i.e. around 10 broadcasts in this example.

7. You are given the following simplified library database schema:

```
Book(BookID, Title, Author)
Patron(PatronID, Name, Address)
Loan(DueDate, PatronID, BookID)
```

(a) Underline the primary keys and dashed-underline all foreign keys.

(b) Write the following queries either in relational algebra or in SQL.

i. Find the names of all patrons who have borrowed at least one book.

ii. Find the names of all authors that patron Jane Smith is currently reading.

(3+3+4)

(b) i: $\pi_{Name}(Patron \bowtie Loan)$

In SQL:

```
SELECT DISTINCT Name
FROM Patron, Loan
WHERE Patron.PatronID = Loan.PatronID
```

ii: $\pi_{Author}(Book \bowtie Loan \bowtie \sigma_{Name='Jane\ Smith'}(Patron))$

In SQL:

```
SELECT DISTINCT Author
FROM Patron, Loan, Book
WHERE Patron.PatronID = Loan.PatronID
AND Loan.BookID = Book.BookID
AND Name = 'Jane Smith'
```