1. Let $A$, $B$, and $C$ be three non-empty sets. Assume that each object either belongs to one set or to all the three sets. That is, there is no object that belongs to two of the sets but not to the third. In symbols:

$$A \cap B \cap C = A \cap B = A \cap C = B \cap C$$

(a) Below is the Venn-diagram for the sets $A$, $B$, and $C$ in which the zones colored with black cannot contain objects.

(b) What is the size of $A \cup B \cup C$ given the following data:

- $|A| = 10$; $|B| = 20$; and $|C| = 30$
- $|A \cap B \cap C| = 5$

**Answer:** The following is implied by the inclusion-exclusion principle for the three sets $A$, $B$, and $C$:

$$|A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |A \cap B \cap C|$$

$$= |A| + |B| + |C| - |A \cap B \cap C| - |A \cap B \cap C| - |A \cap B \cap C| + |A \cap B \cap C|$$

$$= |A| + |B| + |C| - 2|A \cap B \cap C|$$

$$= 10 + 20 + 30 - 2 \cdot 5$$

$$= 50$$

See below the sizes of all the zones in the Venn-diagram for the sets $A$, $B$, and $C$. 
2. (a) **Proposition:** \( \neg(x \wedge y \wedge z) \not\equiv \neg x \wedge \neg y \wedge \neg z \)

**Proof 1:** Consider the assignment \( x = T \), \( y = T \), and \( z = F \). It follows that \( (x \wedge y \wedge z) = F \) and therefore \( \neg(x \wedge y \wedge z) = T \). It also follows that \( (\neg x \wedge \neg y \wedge \neg z) = F \). Therefore, for this assignment, \( \neg(x \wedge y \wedge z) \neq \neg x \wedge \neg y \wedge \neg z \) which implies the proposition.

**Proof 2:** By one of the De Morgan's laws: \( \neg(x \wedge y \wedge z) \equiv \neg x \vee \neg y \vee \neg z \). The proposition follows because the OR boolean function with three variables is equivalent to the AND boolean function with three variables if and only if the three variables are all TRUE or all FALSE. In all other six possible assignments these boolean functions are different.

(b) **Proposition:** \( \neg(x \vee y \vee z) \not\equiv \neg x \vee \neg y \vee \neg z \)

**Proof 1:** Consider the assignment \( x = T \), \( y = F \), and \( z = F \). It follows that \( (x \vee y \vee z) = T \) and therefore \( \neg(x \vee y \vee z) = F \). It also follows that \( (\neg x \vee \neg y \vee \neg z) = T \). Therefore, for this assignment, \( \neg(x \vee y \vee z) \neq \neg x \vee \neg y \vee \neg z \) which implies the proposition.

**Proof 2:** By one of the De Morgan's laws: \( \neg(x \vee y \vee z) \equiv \neg x \wedge \neg y \wedge \neg z \). The proposition follows because the AND boolean function with three variables is equivalent to the OR boolean function with three variables if and only if the three variables are all TRUE or all FALSE. In all other six possible assignments these boolean functions are different.

**Remark:** Note that in the first proof of both part (a) and part (b), there is no need to generate the whole truth table. In order to prove that the two expressions are not equivalent, it is enough to show that the expressions are not the same for at least one assignment.
3. Prove that the following identity is correct by induction on \(n \geq 1\) or by any other method.

\[
\sum_{i=0}^{2n-1} (i + 1) = n(2n + 1)
\]

Note that the sum starts with \(i = 0\) (and not with \(i = 1\)) and ends with \(i = 2n - 1\) (and not with \(i = n\)). As a result, the inductive steps adds two new terms to the sum compared with the number of terms in the sum assumed by the induction hypothesis.

**Proof by induction:**

- **Notations.**
  
  \[
  L(n) = (0 + 1) + (1 + 1) + \cdots + ((2n - 1) + 1) = 1 + 2 + \cdots + 2n \\
  R(n) = n(2n + 1) = 2n^2 + n
  \]

- **Induction base.** Prove that \(L(1) = R(1)\):
  
  \[
  L(1) = 1 + 2 = 3 \\
  R(1) = 2 \cdot 1^2 + 1 = 3
  \]

- **Induction hypothesis.** Assume that \(L(k) = R(k)\) for \(k \geq 1\):
  
  \[
  L(k) = 1 + 2 + \cdots + 2k \\
  R(k) = 2k^2 + k
  \]

- **Inductive step.** Prove that \(L(k + 1) = R(k + 1)\) for \(k \geq 1\):
  
  \[
  L(k + 1) = 1 + 2 + \cdots + 2k + (2k + 1) + (2k + 2) \quad (\ast \text{evaluation of } L(k + 1) \ast) \\
  = L(k) + (2k + 1) + (2k + 2) \quad (\ast \text{definition of } L(k) \ast) \\
  = R(k) + (2k + 1) + (2k + 2) \quad (\ast \text{induction hypothesis} \ast) \\
  = R(k) + (4k + 3) \quad (\ast \text{algebra} \ast) \\
  = (2k^2 + k) + (4k + 3) \quad (\ast \text{evaluation of } R(k) \ast) \\
  = (2k^2 + 4k + 2) + (k + 1) \quad (\ast \text{algebra} \ast) \\
  = 2(k + 1)^2 + (k + 1) \quad (\ast \text{algebra} \ast) \\
  = R(k + 1) \quad (\ast \text{evaluation of } R(k + 1) \ast)
  \]

**Second proof:** Recall that

\[
\sum_{i=1}^{n} i = \frac{n(n+1)}{2}
\]

This identity implies the following,

\[
L(n) = \sum_{i=0}^{2n-1} (i + 1) \quad (\ast \text{definition of } L(n) \ast) \\
= \sum_{i=1}^{2n} i \quad (\ast \text{algebra} \ast) \\
= \frac{2n(2n+1)}{2} \quad (\ast \text{applying the above identity} \ast) \\
= n(2n + 1) \quad (\ast \text{algebra} \ast) \\
= R(n) \quad (\ast \text{definition of } R(n) \ast)
\]
4. Solve the following recursive formula by finding the closed form expression for $T(n)$ as a function of $n$. Assume that $n$ is a positive integer.

$$T(n) = \begin{cases} 
1 & \text{for } n = 1 \\
3 & \text{for } n = 2 \\
2T(n-1) - T(n-2) & \text{for } n > 2 
\end{cases}$$

Small values of $n$:

$$\begin{array}{lcl}
T(1) &=& 1 \\
T(2) &=& 3 \\
T(3) &=& 2 \cdot 3 - 1 = 6 - 1 = 5 \\
T(4) &=& 2 \cdot 5 - 3 = 10 - 3 = 7 \\
T(5) &=& 2 \cdot 7 - 5 = 14 - 5 = 9 \\
T(6) &=& 2 \cdot 9 - 7 = 18 - 7 = 11 
\end{array}$$

Solution: $T(n) = 2n - 1$ for $n \geq 1$.

Proof by induction:

- **Induction base.**
  
  $$\begin{array}{l}
  T(1) = 1 = 2 \cdot 1 - 1 \\
  T(2) = 3 = 2 \cdot 2 - 1 
  \end{array}$$

- **Induction hypothesis.** For $n \geq 3$,
  
  $$\begin{array}{l}
  T(n-1) = 2(n-1) - 1 = 2n - 3 \\
  T(n-2) = 2(n-2) - 1 = 2n - 5 
  \end{array}$$

- **Inductive step.** For $n \geq 3$,
  
  $$\begin{array}{l}
  T(n) = 2T(n-1) - T(n-2) \quad (\ast \text{definition of } T(n) \ast) \\
  = 2(2n - 3) - (2n - 5) \quad (\ast \text{induction hypothesis } \ast) \\
  = 4n - 6 - 2n + 5 \quad (\ast \text{algebra } \ast) \\
  = 2n - 1 \quad (\ast \text{algebra } \ast)
  \end{array}$$
5. Alice, Bob, and Charlie play the following game with a fair coin and a fair 6-face die.

- In the first round they flip the coin. If they flip heads they win and the game ends.
- Otherwise, if they flip tails they throw the dice.
- If they throw 1 or 2 they win and if they throw 3, 4, 5, or 6 they lose and the game ends.

(a) What is the probability that a player throws the dice?

**Answer:** A player throws the dice only if the player flips tails. This happens with probability $\frac{1}{2}$.

(b) What is the winning probability?

**Answer:** When a player flips heads the player wins without throwing the dice. With probability $\frac{1}{2}$ the player throws the dice and then the player wins with probability $\frac{1}{3}$. It follows that the winning probability is

$$\frac{1}{2} + \frac{1}{2} \cdot \frac{1}{3} = \frac{1}{2} + \frac{1}{6} = \frac{2}{3}$$

(c) Bob lost the game. What is the probability that Bob’s dice showed 3?

**Answer:** The only way Bob can lose the game is when Bob flips tails and then throws 3, 4, 5, or 6. Since there are four options to lose the game, the probability that Bob lost the game because the dice showed 3 is $\frac{1}{4}$.

(d) Alice won the game. What is the probability that Alice did not throw the dice?

**Answer:** With probability $\frac{1}{2}$ Alice won the game without throwing the dice and with probability $(\frac{1}{2})(\frac{1}{3}) = \frac{1}{6}$ Alice won after throwing the dice. Therefore, given that Alice won, the probability that Alice did not throw the dice is

$$\frac{\frac{1}{2}}{\frac{1}{2} + \frac{1}{6}} = \frac{3}{4}$$

(e) Charlie replaced the fair dice with a fake one whose six sides show only 1 and 2. What is the probability that Charlie wins the game?

**Answer:** Charlie wins the game with probability 1. This is because Charlie wins either by flipping heads or by throwing the fake dice after flipping tails.

**Some answers using counting arguments:** Assume for simplicity that players always throw the dice even after winning by flipping heads. The following table shows $W$ (win) or $L$ (lose) for each of the twelve possible options of flipping a coin and then throwing a dice.

<table>
<thead>
<tr>
<th>heads</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>tails</td>
<td>W</td>
<td>W</td>
<td>W</td>
<td>W</td>
<td>W</td>
<td>W</td>
</tr>
<tr>
<td>tails</td>
<td>W</td>
<td>W</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>L</td>
</tr>
</tbody>
</table>

(b) There are 12 entries in the table out of which 8 entries are marked with $W$. Therefore, the winning probability is $8/12 = 2/3$.

(c) There are 4 entries that are marked with $L$ out of which only one belongs to the column associated with throwing 3. Therefore, the probability that Bob lost the game because the dice showed 3 is $1/4$.

(d) There are 8 entries that are marked with $W$ out of which 6 entries belong to a win without throwing the dice. Therefore, the probability that Alice won the game without throwing the dice is $6/8 = 3/4$. 
6. Let $P$ be a problem whose input is an array of size $n$ for $n \geq 1$.

- Algorithm $A$ solves $P$ with complexity $\Theta(n)$.
- Algorithm $B$ solves $P$ with complexity $\Theta(\log(n))$.
- Algorithm $C$ solves $P$ with complexity $\Theta(n \log(n))$.
- Algorithm $D$ solves $P$ with complexity $\Theta(n^2)$.
- Algorithm $E$ solves $P$ with complexity $\Theta(1)$.

Order the five algorithms from the least efficient to the most efficient.

**Answer:** The following is the hierarchy among the five functions:

$$1 = o(\log(n)) = o(n) = o(n \log(n)) = o(n^2)$$

Therefore, using $X < Y$ to indicate that algorithm $X$ is more efficient than algorithm $Y$, it follows that the order among the five algorithms from the least efficient one to the most efficient one is as follows:

$$D > C > A > B > E$$

In other words, the least efficient algorithm is Algorithm $D$, the second least efficient algorithm is Algorithm $C$, the third least efficient algorithm is Algorithm $A$, the fourth least efficient algorithm is Algorithm $B$, while the most efficient algorithm is Algorithm $E$. 
7. Below are all the 11 non-isomorphic graphs with 4 vertices.

(a) Mark all the graphs that have at least one cycle.

**Answer:** $G_6, G_8, G_9, G_{10},$ and $G_{11}$.

The first four graphs $G_1, G_2, G_3,$ and $G_4$ each has at most two edges while a cycle must have at least three edges. The other two graphs without a cycle are the trees $G_5$ and $G_7$ and trees do not contain cycles.

(b) Mark all the graphs that are connected.

**Answer:** $G_5, G_7, G_8, G_9, G_{10},$ and $G_{11}$.

All of these graphs are connected because they do not have an isolated vertices or isolated edges. $G_3$ has two isolated edges and $G_1, G_2, G_4,$ and $G_6$ each has at least one isolated vertex.
8. For \( n \geq 2 \), let \( A = [A_1 < A_2 < \cdots < A_n] \) be a sorted array with \( n \) distinct positive integers from the range 1, 2, \ldots, \( n + 1 \). That is, exactly one of the integers from this range is missing in the array \( A \).

**Examples:** The missing integer in the array \([1, 2, 3, 4, 5, 7, 8, 9]\) is 6, the missing integer in the array \([1, 2, 4, 5]\) is 3, the missing integer in the array \([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 14, 15, 16]\) is 12, the missing integer in the array \([2, 3, 4, 5, 6, 7]\) is 1, and the missing integer in the array \([1, 2, 3, 4, 5, 6, 7, 8, 9]\) is 10.

In your own words (use a high-level description and try to avoid codes), describe an efficient algorithm that finds the missing integer. The only questions your algorithm may use are of the type “is \( A_i = i? \)” for some integer \( 1 \leq i \leq n \).

What is the worst-case complexity (number of questions asked) of your algorithm?

**A trivial linear complexity algorithm:** Scan the array by asking if \( A_i = i \) for all indices \( 1 \leq i \leq n \) or until \( A_i \neq i \). The missing integer is \( i \) when \( A_i \neq i \) which implies that \( A_i = i + 1 \). If at the end \( A_n = n \), then the missing integer is \( n + 1 \).

The algorithm is correct because it inspects all the \( n \) indices in \( A \). If the missing integer is \( i < n + 1 \), then it must be found during the scan because in this case \( A[i] = i + 1 \). If the missing integer is \( n + 1 \), then the procedure finishes the scan and returns \( n + 1 \).

The worst-case complexity of the algorithm is \( n \) because when the missing integer is \( n + 1 \) the algorithm needs to examine all the \( n \) entries in the array. When the missing integer is smaller than \( n \) the algorithm asks fewer than \( n \) questions.

**A logarithmic complexity algorithm:** First define \( A_{n+1} = n + 2 \) to make sure that if \( n + 1 \) is the missing integer then \( A_{n+1} \neq n + 1 \) as is the case in which when \( i < n \) is missing then \( A_i \neq i \).

Run the following Binary-Search procedure to find the first index in the array \( A \) for which \( i < A_i \). When the binary-search is asking “is \( A_i = i? \)” for some integer \( 1 \leq i \leq n + 1 \) and the answer is “YES”, then the missing integer must be greater than \( i \). However, if the answer is “NO”, then it must be the case that \( A_i = i + 1 \) and therefore the missing integer must be smaller than \( i \). When the algorithm knows that the missing integer is larger than or equal to \( \ell \) and smaller than or equal to \( r \) for some \( 1 \leq \ell < r \leq n + 1 \), then the next question is about the integer \( m = \lfloor \frac{\ell + r}{2} \rfloor \).

The algorithm is correct because this binary-search like procedure finds the first index \( i \) in \( A \) such that \( A_i = i + 1 \) which always exists after defining \( A_{n+1} = n + 2 \).

The algorithm always asks the same number of questions. This number is \( \lceil \log_2(n + 1) \rceil \) which is the complexity of the binary-search procedure on an array of length \( n + 1 \). Consequently, the complexity of the algorithm is \( \Theta(\log(n)) \).
9. An eight-graph is a graph that is composed of two cycles that share exactly 1 vertex.
   - All the edges of the graph are part of one of the two cycles.
   - Exactly one vertex belongs to both cycles.
   - The rest of the vertices belong only to one of the cycles.
   - A cycle must contain at least 3 vertices including the one that connects both cycles.

An eight-graph must contain at least 5 vertices. See above the only possible eight-graph with 5 vertices and two of the three possible non-isomorphic eight-graphs with 9 vertices. One of them has a cycle of size 4 that is connected to a cycle of size 6 and one of them has two connected cycles of size 5.

(a) Illustrate the only possible non-isomorphic eight-graphs with 6 vertices.

(b) Illustrate the only two possible non-isomorphic eight-graphs with 7 vertices.

(c) For $n \geq 5$, how many non-isomorphic eight-graphs with $n$ vertices exist?
   **Answer:** The size of one of the cycles, denoted by $c_1$, determines the size of the other cycle, denoted by $c_2$. Since the two cycles share exactly one vertex, it follows that $n = c_1 + c_2 - 1$. A cycle must contain at least three vertices and therefore $c_1 \geq 3$ and $c_2 \geq 3$ which implies that $c_1 \leq n - 2$. As a result, there are exactly $n - 4$ possible values for $c_1$ taken from the range 3, 4, ..., $n - 2$. However, the eight-graph in which the size of one cycle is $c$ and the size of the other cycle is $n + 1 - c$ is counted twice once when $c_1 = c$ and once when $c_1 = n + 1 - c$. It follows that for an even $n$ the $n/2 - 2$ non-isomorphic sizes for $c_1$ are 3, 4, ..., $n/2$ and for an odd $n$ the $(n+1)/2 - 2$ non-isomorphic sizes for $c_1$ are 3, 4, ..., $(n+1)/2$.

(d) For $n \geq 5$, all possible eight-graphs have the same number of edges. What is this number as a function of $n$?
   **Answer:** Denote by $c_1$ and $c_2$ the sizes of the two cycles. In the previous part, it was shown that $n = c_1 + c_2 - 1$. A cycle with $c$ vertices has $c$ edges. The edges of both cycles are disjoint. Hence, the number of edges $m$ of an eight-graph with $n$ vertices is $c_1 + c_2$. That is, $m = n + 1$ in an eight-graph.

(e) For $n \geq 5$, the degree sequence of all possible eight-graphs is the same. Let $G$ be an eight-graph with $n$ vertices. How many vertices in $G$ have degree 0? How many vertices in $G$ have degree 1? How many vertices in $G$ have degree 2? How many vertices in $G$ have degree 3? How many vertices in $G$ have degree 4? How many vertices in $G$ have degree greater than 4?
   **Answer:** Let $v$ be the vertex shared by the two cycles. It follows that the degree of $v$ is 4 while the degrees of any of the other $n - 1$ vertices in the cycles are 2. Therefore, an eight-graph with $n$ vertices has zero vertices with degree 0 or 1, $n - 1$ vertices with degree 2, zero vertices with degree 3, one vertex with degree 4. None of the vertices have degree greater than 4.