Problem 1

Use induction to prove that:

\[
\left(1 - \frac{1}{4}\right) \left(1 - \frac{1}{9}\right) \ldots \left(1 - \frac{1}{n^2}\right) = \frac{n + 1}{2n}
\]

For all \(n \geq 2\).

Problem 2

**Conjunctive Normal Form** (CNF) is a particular way of writing Boolean formulae that will be used in this course. We define it constructively:

1. A *literal* is a variable or its complement. \(x_i\) and \(\neg x_i\) are examples of literals.
2. A *clause* is any disjunction (that is, OR) of any number of literals. A literal by itself is also a clause.
3. A formula is *in conjunctive normal form* if it is the conjunction (that is, AND) of any number of clauses.

As an example, the formula \((x_1 \lor x_3) \land (\neg x_1 \lor x_2) \land \neg x_2\) is in CNF.
Using the above definition, convert the following formulae into conjunctive normal form:

a. \((x_1 \land x_2) \land \neg(x_2 \land x_3) \land (x_1 \lor x_4)\)

b. \((x_1 \land x_2) \land (\neg(x_2 \land x_3) \lor (x_1 \lor x_4))\)

c. \((x_1 \oplus x_2) \rightarrow (x_2 \oplus x_3)\)

d. \(((x_1 \leftrightarrow (\neg x_2 \land x_3)) \leftrightarrow x_2)\)

Problem 3

Suppose \(G = (V, E)\) is an undirected graph in which every vertex has degree at least \(\frac{|V| - 1}{2}\). Prove that \(G\) is connected.

The following questions are lab problems.

Lab Problem 1

1. When it was first discovered that there are computational problems that cannot be solved by computers, it was a very surprising result. In this problem, we will do a high-level argument of a related result, due to Gödel. Gödel showed that there exist true mathematical statements that cannot be proven. Instead of using mathematical statements, we will simply use English sentences.

This is the sentence we are trying to prove:

\textit{There exists a true sentence for which there is no proof.}

We will argue this by contradiction.\footnote{We are being informal here, because we have not given a mathematical formalization of the statement we are trying to prove.} That is, we will assume the following statement, then obtain a contradiction:

\textit{Assumption: All true sentences have proofs.}
You may assume that any statement for which there exists a proof is true. (Hint: Examine the sentence $S = \text{“No proof exists for this sentence.”}$ Show that given our assumption, if $S$ is true or false, we obtain a contradiction.)

2. In class we looked at the Halting Problem, the problem of determining whether a program $P$ will terminate on an input $X$. We wanted to know whether a program $\text{HPSolver}$ solving the Halting Problem exists:

$\text{HPSolver:}$
- **Input:** a program $P$, an input $X$
- **Output:** $\text{Yes}$ if $P$ terminates on $X$, $\text{No}$ if $P$ runs forever

We concluded that $\text{HPSolver}$ cannot exist, meaning that no program can solve the halting problem. We will now look at another problem that cannot be solved by computers, the problem of deciding the truth or falsity of a sentence. We want to know whether a program $\text{TruthSolver}$ exists:

$\text{TruthSolver:}$
- **Input:** a sentence $S$
- **Output:** $\text{True}$ if $S$ is true, $\text{False}$ if $S$ is false

Argue that $\text{TruthSolver}$ cannot exist, by first showing that if we had a $\text{TruthSolver}$, we could use it to build an $\text{HPSolver}$. Then argue that since $\text{HPSolver}$ cannot exist, $\text{TruthSolver}$ cannot exist. (Hint: $\text{HPSolver}$ will use $\text{TruthSolver}$ as a subroutine. What kind of statement should it give as input?)

**Lab Problem 2**

**Mealy vs. Moore** Moore machines are finite state machines (FSMs) where an output/action is produced when the FSM reaches a state. That is to say, each transition is labeled by some input symbol, and each state is labeled with an output/action. When the machine enters a state, it produces the corresponding output/action.

In a Mealy machine an output/action is produced as the FSM makes a transition from one state to another, not after it reaches a state. That is, each transition is labeled with a pair $\langle\text{input}\rangle/\langle\text{action}\rangle$, and the action is performed as the machine transitions from one state to another.
a) Given this Mealy machine, give us an equivalent Moore machine, one that will produce the same action/output on the same input.

b) Given this Moore machine, give us an equivalent Mealy machine, one that will produce the same action/output on the same input.

c) It turns out that, in general, for every Mealy machine there is an equivalent Moore machine, and vice versa. You just saw an example
of this equivalence. Based on the way you solved (a) and (b), explain a systematic way of constructing a corresponding Moore machine from a Mealy machine, and vice versa.

Lab Problem 3

Suppose \( n \) chords are drawn into a circle, dividing it into regions. Show that the regions can always be colored using two colors so that adjacent regions (i.e. regions that share an edge) have different colors (as in the example below).