1. a. Let 2-Satisfying Assignments be the following problem.
Input: A CNF formula F (i.e., a boolean formula which is an “and” of clauses, where each
clause is an “or” of boolean variables and their complements).
Question: Does F have two distinct satisfying assignments?
e.g. $F_1 = x_1 \lor x_2$ has the satisfying assignments $x_1 = \text{TRUE}, x_2 = \text{FALSE}$ and $x_1 = \text{FALSE}, x_2 = \text{TRUE}$; $F_2 = (x_1 \lor x_2) \land (\overline{x}_1)$ has just one satisfying assignment, namely
$x_1 = \text{FALSE}, x_2 = \text{TRUE}$.
Show that 2-Satisfying Assignments has a polynomial time verifier. That is, given a
“certificate” $C$ of size polynomial in $n$, where $n$ is the input size, there is a polynomial
time algorithm to check whether $C$ certifies that the input $F$ is in the language 2-Satisfying
Assignment, such that for each $F \in$ 2-Satisfying Assignments, there is a polynomial-sized
certifying $C$.

b. Let Subset Sum be the following problem.
Input: A collection of $n$ not necessarily distinct integers and a target integer $t$.
Question: Is there a subset of the collection, such that the numbers in the subset sum to
exactly $t$?
Show that Subset Sum has a polynomial time verifier.

2. Suppose that you were given a polynomial time algorithm for Satisfiability, that is an
algorithm that reports whether or not the input is a satisfiable Boolean formula.
Use it to give a polynomial time algorithm to find a satisfying assignment if the formula
is satisfiable.
Hint. Let $x_1, x_2, \ldots, x_m$ be the variables in the formula. First determine if there is a sat-
isfying assignment with $x_1 = \text{TRUE}$. Now try and find a complete satisfying assignment.
Note, however, that even if each of $x_1 = \text{TRUE}$ and $x_2 = \text{TRUE}$ occur in satisfying as-
signments, it need not be the case that they both occur in a single satisfying assignment
(e.g. $F = (x_1 \lor \overline{x}_2) \land (\overline{x}_1 \lor x_2)$).

3. Suppose that you were given a polynomial time algorithm for Satisfiability. Use it to give
a polynomial algorithm for Clique.
That is, given a graph $G$ and an integer $k$, the task is to give an algorithm to determine if
$G$ has a clique of size $k$. To do this, you will use the Satisfiability algorithm as a subroutine.
To this end, you will construct a Boolean formula $F_{G,k}$ such that $F_{G,k}$ is satisfiable if and only
if $G$ has a clique of size $k$. The main task is to describe how to construct $F_{G,k}$ in polynomial
time.

4. Let Equal Subset Sum be the following problem.
Input: A collection of $n$ not necessarily distinct integers, $a_1, a_2, \cdots, a_n$.
Question: Is there a subset of the collection, such that the numbers in the subset sum to
exactly $\frac{1}{2} \sum_{i=1}^{n} a_i$?
Suppose that you were given a polynomial time algorithm for Equal Subset Sum. Use it
as a subroutine to give a polynomial time algorithm for Subset Sum.