1 Formal Specification

For each informal description, give a formal specification as a program type, a precondition and a postcondition. Ambiguities in the informal description can be resolved in any way consistent with standard English usage.

1. Given an equation of the form $f(x) = 0$, where $f$ is given and $x$ is the unknown, the program computes the smallest solution to the equation.  

2. The program decomposes a number into the sum of four squares. For instance, $671 = 1^2 + 2^2 + 15^2 + 21^2$ and $645 = 0^2 + 2^2 + 4^2 + 25^2$.  

3. Given an array $A[0..N-1]$ of positive integers and a target $x$, the program finds a list of numbers from $A$ that add up to $x$, if such a list exists. Note that no number from $A$ can be used more than once.  


5. Twin primes are prime numbers that differ by exactly two. For instance, 3 and 5 are twin primes and so are 659 and 661. The program calculates the first $N$ pairs of twin primes. You can assume that a predicate isPrime has been defined and you can use it in the specification.

2 Program Verification

1. Consider the following triple for partial correctness:  

$\{P\}$  

$S_0;$  

while $C_0$ do  

while $C_1$ do  

$S_1;$ $S_2$  

done;  

if $C_2$ then $S_3$ else $S_4$ end  

done  

$\{Q\}$  

Write all the numbered proof obligations to be discharged to solve this problem. Pick names for all intermediate predicates (loop invariants and "middle predicates" for sequential composition) and use a consistent numbering when splitting a proof obligation into more elementary subgoals (e.g., proof obligation $n$ results into proof obligations $n.1$ and $n.2$). **Hint:** There are 14 proof obligations, numbered $(1, 2, \ldots, 2.3.1.3.2)$, 9 of which are elementary and cannot be further decomposed (i.e., the leaves of the proof tree).
2. Prove the following triple for partial correctness: 50 pts

{\(N > 0\)}

if \(A[0] > 0\) then
   \(r := true\)
else
   \(r := false; i := 1;\)
   while \(i < N\) do
      if \(A[i] > 0\) then \(r := true\) end;
      \(i := i + 1\)
   end
end

{\(r \equiv \exists x \in 0..N - 1 (A[x] > 0)\)}

Be sure to indicate clearly what your loop invariant is and to write explicitly all proof obligations, using a consistent numbering scheme. Provide a detailed proof for those obligations that are not trivially true. You can omit details from proofs that are generic (i.e., independent from this particular program), like for instance the fact that \(n.m\) follows from \(n.m.1\) and \(n.m.2\) using theorem (8.14b).
### Table 6.1

<table>
<thead>
<tr>
<th>Negation</th>
<th>Disjunction</th>
<th>Conjunction</th>
<th>Implication</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\neg \neg A \equiv A$</td>
<td>$A \lor true \equiv true$</td>
<td>$A \land true \equiv A$</td>
<td>$A \implies true \equiv true$</td>
</tr>
<tr>
<td>$A \lor false \equiv A$</td>
<td>$A \land false \equiv false$</td>
<td>$A \land A \equiv A$</td>
<td>$A \implies false \equiv \neg A$</td>
</tr>
<tr>
<td>$A \land false \equiv false$</td>
<td>$A \land A \equiv A$</td>
<td>$true \implies A \equiv A$</td>
<td>$false \implies A \equiv true$</td>
</tr>
<tr>
<td>$A \lor \neg A \equiv true$</td>
<td>$A \land false \equiv false$</td>
<td>$false \implies A \equiv true$</td>
<td>$A \implies A \equiv true$</td>
</tr>
</tbody>
</table>

### Some Conversions

<table>
<thead>
<tr>
<th>Absorption Laws</th>
</tr>
</thead>
</table>

### De Morgan's Laws

$A \land (A \lor B) \equiv A \lor \neg (A \land B)$

$A \lor (A \land B) \equiv A \land \neg (A \lor B)$

$A \land (A \lor B) \equiv A \lor \neg (A \land B)$

$A \lor (A \land B) \equiv A \land \neg (A \lor B)$

### Equivalence

$(A \equiv B) \equiv (A \implies B) \land (B \implies A)$

### Inference Rules for Propositions

**Conjunction (Conj)**

\[
\frac{A, B}{\therefore A \land B}
\]

**Simplification (Simp)**

\[
\frac{A \land B}{\therefore A}
\]

**Addition (Add)**

\[
\frac{A}{\therefore A \lor B}
\]

**Disjunctive Syllogism (DS)**

\[
\frac{A \lor B, \neg A}{\therefore B}
\]

**Hypothetical Syllogism (HS)**

\[
\frac{A \implies B, B \implies C}{\therefore A \implies C}
\]

**Modus Ponens (MP)**

\[
\frac{A \implies B, A}{\therefore B}
\]

**Modus Tollens (MT)**

\[
\frac{A \implies B, \neg B}{\therefore \neg A}
\]

**Constructive Dilemma (CD)**

\[
\frac{A \lor B, A \implies C, B \implies D}{\therefore C \lor D}
\]

**Destructive Dilemma (DD)**

\[
\frac{\neg C \lor \neg D, A \implies C, B \implies D}{\therefore \neg A \lor \neg B}
\]
C  Inference Rules for Quantifiers

- Renaming: if there are no free occurrences of \( y \) in \( W \),
  \[
  \forall x W \equiv \forall y W(x/y) \quad \text{and} \quad \exists x W \equiv \exists y W(x/y)
  \]

- Universal Instantiation (UI):
  \[
  \forall x W \quad \therefore W(x/t)
  \]

- Existential Generalization (EG):
  \[
  W(x/t) \quad \therefore \exists x W
  \]

- Existential Instantiation (EI): if \( c \) is a new constant in the proof and \( c \) does not occur in the conclusion of the proof:
  \[
  \exists x W \quad \therefore W(x/c)
  \]

- Universal Generalization (UG): if among the WFFS used to infer \( W \), \( x \) is not free in any premise and \( x \) is not free in any WFF constructed by EI:
  \[
  W \quad \therefore \forall x W
  \]

D  Known Predicates

D.1  Set Notation

\[
\forall x \in D p(x) \equiv \forall x((x \in d) \implies p(x))
\]
\[
\exists x \in D p(x) \equiv \exists x((x \in d) \land p(x))
\]

D.2  Implications and Equivalences

\[
\forall x A(x) \implies \exists x A(x) \tag{7.2a}
\]
\[
\exists x (A(x) \land B(x)) \implies \exists x A(x) \land \exists x B(x) \tag{7.2b}
\]
\[
\forall x A(x) \lor \forall x B(x) \implies \forall x (A(x) \lor B(x)) \tag{7.2c}
\]
\[
\forall x (A(x) \rightarrow B(x)) \implies (\forall x A(x) \implies \forall x B(x)) \tag{7.2d}
\]
\[
\exists y \forall x P(x, y) \implies \forall x \exists y P(x, y) \tag{7.2e}
\]
\[
\forall x A(x) \implies \exists x A(x) \implies A(x) \lor \exists x B(x) \tag{7.6}
\]
\[
\forall x (A(x) \land B(x)) \equiv \forall x A(x) \land \exists x B(x) \tag{7.7}
\]

D.3  Restricted Equivalences

If there are no free occurrences of \( x \) in \( C \):

\[
\forall x C \equiv C \tag{7.9}
\]
\[
\exists x C \equiv C \tag{7.10}
\]
\[
\forall x (C \lor A(x)) \equiv C \lor \forall x A(x) \tag{7.10}
\]
\[
\exists x (C \lor A(x)) \equiv C \lor \exists x A(x) \tag{7.10}
\]
\[
\forall x (C \land A(x)) \equiv C \land \forall x A(x) \tag{7.11}
\]
\[
\exists x (C \land A(x)) \equiv C \land \exists x A(x) \tag{7.11}
\]
\[
\forall x (A(x) \rightarrow C) \equiv \exists x A(x) \rightarrow C \tag{7.12}
\]
\[
\forall x (A(x) \rightarrow C) \equiv \forall x A(x) \rightarrow C \tag{7.12}
\]
\[
\forall x (A(x) \rightarrow C) \equiv \exists x A(x) \rightarrow C \tag{7.12}
\]
\[
\forall x (A(x) \rightarrow C) \equiv \forall x A(x) \rightarrow C \tag{7.12}
\]
E  Inference Rules and Theorems for Partial Correctness

- skip Axiom:
  \[
  \{Q\} \text{skip} \{Q\} \tag{skip}
  \]

- Assignment Axiom:
  \[
  \{Q(x/t)\} \ x := t \ {Q} \tag{8.10}, (AA)
  \]

- Consequence Rules:
  \[
  P \implies R, \ \{R\} \ S \ \{Q\} \quad \therefore \quad \{P\} \ S \ \{Q\} \tag{8.11a}
  \]
  \[
  \{P\} \ S \ \{T\}, \ T \implies Q \quad \therefore \quad \{P\} \ S \ \{Q\} \tag{8.11b}
  \]

- Composition Rule:
  \[
  \{P\} \ S_1 \ \{R\}, \ \{R\} \ S_2 \ \{Q\} \quad \therefore \quad \{P\} \ S_1; \ S_2 \ \{Q\} \tag{8.12}
  \]

- Composition of assignments:
  \[
  \{Q(y/u)(x/t)\} \ x := t; \ y := u \ {Q} \tag{CA}
  \]

- If-Then-Else Rule:
  \[
  \{P \land C\} \ S_1 \ \{Q\}, \ \{P \land \neg C\} \ S_2 \ \{Q\} \quad \therefore \quad \{P\} \ \text{if} \ C \ \text{then} \ S_1 \ \text{else} \ S_2 \ \text{end} \ \{Q\} \tag{8.14a}
  \]

- If-Then Rule:
  \[
  \{P \land C\} \ S \ \{Q\}, \ P \land \neg C \implies Q \quad \therefore \quad \{P\} \ \text{if} \ C \ \text{then} \ S \ \text{end} \ \{Q\} \tag{8.14b}
  \]

- While Rule:
  \[
  \{I \land C\} \ S \ \{I\} \quad \therefore \quad \{I\} \ \text{while} \ C \ \text{do} \ S \ \text{end} \ \{I \land \neg C\}_p \tag{8.15a}
  \]

- Conjunction Rule:
  \[
  \{P\} \ S \ \{Q\}, \ \{P'\} \ S \ \{Q'\} \quad \therefore \quad \{P \land P'\} \ S \ \{Q \land Q'\} \tag{Conj}
  \]

- Disjunction Rule:
  \[
  \{P\} \ S \ \{Q\}, \ \{P'\} \ S \ \{Q'\} \quad \therefore \quad \{P \lor P'\} \ S \ \{Q \lor Q'\} \tag{Disj}
  \]

F  While Rule for Total Correctness

\[
\begin{align*}
&\{I \land C\} \ S \ \{I\}_t, \\
&\forall k \in \mathbb{N} \ \{I \land C \land (V = k)\} \ S \ \{V < k\}_t, \\
&I \land C \implies V \geq 0 \\
&\therefore \quad \{I\} \ \text{while} \ C \ \text{do} \ S \ \text{end} \ \{I \land \neg C\}_t \tag{8.15b}
\end{align*}
\]