Exercises for which solution notes are available

**Exercise 1**
Write a specification which is true if and only if the following program terminates.

\[
\text{WHILE } X > 1 \text{ DO IF ODD}(X) \text{ THEN } X := (3 \times X) + 1 \text{ ELSE } X := X \div 2
\]

**Exercise 2**
Let \( C \) be the following command

\[
R := X; \\
Q := 0; \\
\text{WHILE } Y \leq R \text{ DO (} R := R - Y; \ Q := Q + 1 \text{)}
\]

Find a condition \( P \) such that \([P] \ C \ [R < Y \land X = R + (Y \times Q)]\) is true.

**Exercise 3**
When is \([T] \ C \ [T]\) true?

**Exercise 4**
Write a partial correctness specification which is true if and only if the command \( C \) has the effect of multiplying the values of \( X \) and \( Y \) and storing the result in \( X \).

**Exercise 5**
Write a specification which is true if the execution of \( C \) always halts when execution is started in a state satisfying \( P \).

**Exercise 6**
Find the flaw in the ‘proof’ of \( 1 = -1 \) below:

1. \( \sqrt{-1 \times -1} = \sqrt{-1} \times \sqrt{-1} \) Reflexivity of \( = \).
2. \( \sqrt{-1 \times -1} = (\sqrt{-1}) \times (\sqrt{-1}) \) Distributive law of \( \sqrt{\cdot} \) over \( \times \).
3. \( \sqrt{-1 \times -1} = (\sqrt{-1})^2 \) Definition of \( (\cdot)^2 \).
4. \( \sqrt{-1} \times \sqrt{-1} = -1 \) definition of \( \sqrt{-1} \).
5. \( \sqrt{1} = -1 \) As \(-1 \times -1 = 1\).
6. \( 1 = -1 \) As \( \sqrt{1} = 1 \).

**Exercise 7**
Is the following specification true?

\( \vdash \{ X = x \land Y = y \} : X := X + Y; \ Y := X - Y; \ X := X - Y \ \{ Y = x \land X = y \} \)

If so, prove it. If not, give the circumstances in which it fails.
Exercise 8
Show in detail that \( \vdash \{ X = R + (Y \times Q) \} \quad R := R - Y; \quad Q := Q + 1 \quad \{ X = R + (Y \times Q) \} \)

Exercise 9
Give a detailed formal proof that
\( \vdash \{ T \} \quad \text{IF } X \geq Y \text{ THEN } \text{MAX} := X \text{ ELSE } \text{MAX} := Y \quad \{ \text{MAX} = \max(X, Y) \} \)
follows from \( \vdash X \geq Y \Rightarrow \max(X, Y) = X \) and \( \vdash Y \geq X \Rightarrow \max(X, Y) = Y. \)

Exercise 10
Suppose we add to our little programming language commands of the form:

\[
\text{CASE } E \text{ OF BEGIN } C_1; \ldots ; C_n \text{ END}
\]

These are evaluated as follows:

(i) First \( E \) is evaluated to get a value \( x \).

(ii) If \( x \) is not a number between 1 and \( n \), then the \text{CASE}-command has no effect.

(iii) If \( x = i \) where \( 1 \leq i \leq n \), then command \( C_i \) is executed.

Why is the following rule for \text{CASE}-commands wrong?

\[
\begin{align*}
\vdash & \{ P \land E = 1 \} \quad C_1 \quad \{ Q \}, \ldots , \vdash \{ P \land E = n \} \quad C_n \quad \{ Q \} \\
\vdash \ & P \quad \text{CASE } E \text{ OF BEGIN } C_1; \ldots ; C_n \text{ END } \{ Q \}
\end{align*}
\]

Hint: Consider the case when \( P \) is ‘\( X = 0 \)’, \( E \) is ‘\( X \)’, \( C_1 \) is ‘\( Y := 0 \)’ and \( Q \) is ‘\( Y = 0 \)’.

Exercise 11
Devise a proof rule for the \text{CASE}-commands in the previous exercise and use it to show:

\[
\vdash \{ 1 \leq X \land X \leq 3 \} \quad \text{CASE } X \text{ OF BEGIN } Y := X-1; Y := X-2; Y := X-3 \text{ END } \{ Y = 0 \}
\]

Exercise 12
Devise a proof rule for a command

\[
\text{REPEAT } \text{command} \text{ UNTIL } \text{statement}
\]
The meaning of \texttt{REPEAT C UNTIL S} is that \( C \) is executed and then \( S \) is tested; if the result is true, then nothing more is done, otherwise the whole \texttt{REPEAT} command is repeated. Thus \texttt{REPEAT C UNTIL S} is equivalent to \( C; \texttt{WHILE} \neg S \texttt{ DO C} \).

\textbf{Exercise 13}

Show that
\[ \vdash \{ M \geq 1 \}
\begin{align*}
X &:= 0; \\
\text{FOR } N := 1 \text{ UNTIL } M \text{ DO } X := X + N \\
\{ X = (M \times (M+1)) \text{ DIV } 2 \} 
\end{align*} \]

\textbf{Exercise 14}

Show
\[ \vdash \{ A(X) = x \land A(Y) = y \land X \neq Y \}
\begin{align*}
A(X) &:= A(X) + A(Y); \\
A(Y) &:= A(X) - A(Y); \\
A(X) &:= A(X) - A(Y) \\
\{ A(X) = y \land A(Y) = x \} 
\end{align*} \]

Why is the precondition \( X \neq Y \) necessary?

\textbf{Exercise 15}

Prove
\[ \vdash \{ 1 \leq N \}
\begin{align*}
\text{FOR } I := 1 \text{ UNTIL } N \text{ DO } A(I) := 0 \\
\{ \text{SORTED}(A,N) \} 
\end{align*} \]

\textbf{Exercise 16}

Prove
\[ \vdash \{ 1 \leq N \land A = a \}
\begin{align*}
N &:= 1 \\
\{ \text{SORTED}(A,N) \land \text{PERM}(A,a,N) \} 
\end{align*} \]
Additional exercises without solution notes

Exercise 17
Use your REPEAT rule to deduce:

\[ \begin{align*}
\vdash \{ S = C+R \land R < Y \} \\
& \text{REPEAT } (S:=S+1; \ R:=R+1) \text{ UNTIL } R = Y \\
& \{ S = C+Y \}
\end{align*} \]

Exercise 18
Use your REPEAT rule to deduce:

\[ \begin{align*}
\vdash \{ X=x \land Y=y \} \\
& S:=0; \\
& \text{REPEAT} \\
& \quad R:=0; \\
& \quad \text{REPEAT } (S:=S+1; \ R:=R+1) \text{ UNTIL } R = Y; \\
& \quad X:=X-1 \\
& \quad \text{UNTIL } X = 0 \\
& \{ S = x \times y \}
\end{align*} \]

Exercise 19
The exponentiation function \( \text{exp} \) satisfies:

\[ \begin{align*}
\text{exp}(m, 0) & = 1 \\
\text{exp}(m, n+1) & = m \times \text{exp}(m, n)
\end{align*} \]

Devise a command \( C \) that uses repeated multiplication to achieve the following partial correctness specification:

\[ \begin{align*}
\{ X=x \land Y=y \land Y \geq 0 \} \ C \ { \{ Z=\text{exp}(x, y) \land X=x \land Y=y \} }
\end{align*} \]

Prove that your command \( C \) meets this specification.

Exercise 20
Assume \( \text{gcd}(X,Y) \) satisfies:

\[ \begin{align*}
\vdash (X > Y) & \Rightarrow \text{gcd}(X,Y) = \text{gcd}(X-Y,Y) \\
\vdash \text{gcd}(X,Y) & = \text{gcd}(Y,X) \\
\vdash \text{gcd}(X,X) & = X
\end{align*} \]

Prove:

\[ \begin{align*}
\vdash \{ (A > 0) \land (B > 0) \land (\text{gcd}(A,B) = \text{gcd}(X,Y)) \} \\
& \text{WHILE } A > B \text{ DO } A := A - B; \\
& \text{WHILE } B > A \text{ DO } B := B - A \\
& \{ (0 < B) \land (B \leq A) \land (\text{gcd}(A,B) = \text{gcd}(X,Y)) \}
\end{align*} \]
Hence, or otherwise, use your rule for REPEAT commands to prove:

\[ \{ A=a \land B=b \} \]

\[ \text{REPEAT} \]
\[ \text{WHILE} A>B \text{ DO } A:=A-B; \]
\[ \text{WHILE} B>A \text{ DO } B:=B-A \]
\[ \text{UNTIL} A=B \]
\[ \{ A=B \land A=gcd(a,b) \} \]

Exercise 21
Deduce:

\[ \{ S = (x\times y)-(X\times Y) \} \]

\[ \text{WHILE } \neg \text{ODD}(X) \text{ DO } (Y:=2\times Y; X:=X \div 2) \]
\[ \{ S = (x\times y)-(X\times Y) \land \text{ODD}(X) \} \]

Exercise 22
Deduce:

\[ \{ S = (x\times y)-(X\times Y) \} \]

\[ \text{WHILE } \neg (X=0) \text{ DO } \]
\[ \text{WHILE } \neg \text{ODD}(X) \text{ DO } (Y:=2\times Y; X:=X \div 2); \]
\[ S:=S+Y; \]
\[ X:=X-1 \]
\[ \{ S = x\times y \} \]

Exercise 23
Deduce:

\[ \{ X=x \land Y=y \} \]

\[ S:=0; \]
\[ \text{WHILE } \neg (X=0) \text{ DO } \]
\[ (\text{WHILE } \neg \text{ODD}(X) \text{ DO } (Y:=2\times Y; X:=X \div 2); \]
\[ S:=S+Y; \]
\[ X:=X-1) \]
\[ \{ S = x\times y \} \]

Exercise 24
Using \( P \times X^N = x^n \) as an invariant, deduce:

\[ \{ X=x \land N=n \} \]

\[ P:=1; \]
\[ \text{WHILE } \neg (N=0) \text{ DO } \]
\[ (\text{IF ODD}(N) \text{ THEN } P:=P \times X \text{ else } P:=P; \]
\[ N:=N \div 2; \]
\[ X:=X \times X) \]
\[ \{ P = x^n \} \]
**Exercise 25**
Prove that the command

\[
Z := 0; \\
\text{WHILE } \neg (X = 0) \text{ DO} \\
\text{IF } \text{ODD}(X) \text{ THEN } Z := Z + Y \text{ ELSE } Z := Z; \\
Y := Y \times 2; \\
X := X \div 2)
\]
computes the product of the initial values of \(X\) and \(Y\) and leaves the result in \(Z\).

**Exercise 26**
Prove that the command

\[
Z := 1; \\
\text{WHILE } N > 0 \text{ DO} \\
\text{IF } \text{ODD}(N) \text{ THEN } Z := Z \times X \text{ else } Z := Z; \\
N := N \div 2; \\
X := X \times X)
\]
assigns \(x^n\) to \(Z\), where \(x\) and \(n\) are the initial values of \(X\) and \(N\) respectively and we assume \(n \geq 0\).

**Exercise 27**
What are the verification conditions for the following specification?

\[
\{T\} \text{ IF } X \geq Y \text{ THEN } \text{MAX} := X \text{ ELSE } \text{MAX} := Y \{\text{MAX} = \text{max}(X,Y)\}
\]
Are they true?

**Exercise 28**
What are the verification conditions for the following specification?

\[
\{X = R + (Y \times Q)\} \ R := R - Y; \ Q := Q + 1 \{X = R + (Y \times Q)\}
\]
Are they true?

**Exercise 29**
What are the verification conditions generated by the following annotated specification. Are they true?

\[
\{X=n\} \\
Y := 1; \ \{Y = 1 \land X = n\} \\
\text{WHILE } X \neq 0 \text{ DO} \{Y \times X! = n!\} \\
\text{(Y := Y \times X; X := X - 1)} \\
\{X = 0 \land Y = n!\}
\]
Exercise 30
Why are the verification conditions for the annotated specification

{\top} \text{WHILE } F \text{ DO } \{F\} \text{ } X:=0 \text{ } \{\top\}

not provable, even though $\vdash \{\top\} \text{WHILE } F \text{ DO } X:=0 \text{ } \{\top\}$.

Exercise 31
Prove by induction on the structure of $C$ that if no variable occurring in $P$ is assigned to in $C$, then $\vdash \{P\} C \{P\}$.

Exercise 32
Devise verification conditions for commands of the form $\text{REPEAT } C \text{ UNTIL } S$ (see Exercise 12).

Exercise 33
Consider the following alternative scheme for generating VCs from annotated \textsc{WHILE}-commands (due to Silas Brown).

\begin{center}
\begin{tabular}{|c|}
\hline
\textbf{\textsc{WHILE}-commands} \\
\hline
Alternative verification conditions generated from \\
$\{P\} \text{WHILE } S \text{ DO } \{R\} C \{Q\}$ \\
\hline
are \\
(i) $P \land S \Rightarrow R$ \\
(ii) $P \land \neg S \Rightarrow Q$ \\
(iii) the verification conditions generated by \\
$\{R\} C\{(Q \land \neg S) \lor (R \land S)\}$ \\
\hline
\end{tabular}
\end{center}

Either justify these VCs, or find a counterexample.