1. For each of the following, either give a short (3 lines max.) justification or give a counterexample.

(a) Let \( f : \mathbb{N} \to \mathbb{N} \) be a total function with the property that for every primitive recursive function \( g : \mathbb{N} \to \mathbb{N} \) there exists an \( N \in \mathbb{N} \) with \( f(x) > g(x) \) for all \( x > N \). Then \( f \) is not recursive.

(b) If \( f : \mathbb{N} \to \mathbb{N} \) has a finite domain, then \( f \) is partial recursive.

(c) If \( f : \mathbb{N} \to \mathbb{N} \) is primitive recursive, then \( Gr(f) = \{(x, f(x)) \mid x \in \mathbb{N}\} \) is also primitive recursive.

---

a) The Ackermann function \( A(x, y) \) is recursive but not PR (it dominates each PR function). So \( A'(x) = A(x, x) \) is recursive and dominates all PR functions.

b) True, if \( \text{dom}(f) = \{a_1, \ldots, a_k\} \) we have
\[
 f(x) = \begin{cases} 
 f(a_i) & \text{if } x = a_i \\
 f(a_k) & \text{if } x = a_k \\
 0 & \text{o.w.} 
\end{cases}
\]

---

c) \( \chi_{\text{dom}}(x, y) = \begin{cases} 
 1 & \text{if } y = f(x) \\
 0 & \text{o.w.} 
\end{cases} \)

Since \( f \), \( = \), and \( \chi_{\text{dom}}(x, y) \) are PR, so is \( \chi_{\text{dom}}(g) \).
2. For a natural number $n$, set $f(n) = n/2$ if $n$ is even, and $f(n) = 3n+1$ if $n$ is odd. The $3n+1$ problem (an open problem in mathematics) asks whether for every $n > 0$, the sequence $n, f(n), f(f(n)), \ldots, f^{(k)}(n), \ldots$ eventually reaches 1.

Write a register machine program that, on input $n$, checks whether $f^{(k)}(n) = 1$ for some $k$. If such $k$ exists, the program should output 1. If not, the program should not halt.

You are allowed to use the following macros:

- Even($i$), which sets register $i$ to 1 if $R_i$ is even, and to 0 otherwise.
- Odd($i$), which sets register $i$ to 1 if $R_i$ is odd, and to 0 otherwise.
- Mult($i, j, k$), which sets $R_k := R_iR_j$. (Multiply the values of $R_i$ and $R_j$ and place the result in $R_k$.)
- Half($i$), which sets $R_i := R_i/2$ for $R_i$ even and leaves $R_i$ unchanged for $R_i$ odd.

You may also use the macros from Table 2.1 in the notes. Indicate in words how your program works.

\begin{verbatim}
  1. Inc 4
  2. Inc 4
  3. Inc 4
  4. Inc 3
  5. Jump (1,3,13)  \hspace{1cm} \text{If } R_1=1, \text{ then done}
  6. Copy (1,2)
  7. Odd 2
  8. Half 1
  9. Jump (2,5)
  10. Mult (1,4,1)
 11. Inc 1
 12. Jump 5
 13. Inc (0)
\end{verbatim}

\begin{verbatim}
  R_{n+1} = R_n \mod 2
  5-9 \text{ removes factors of 2 from } R_n
  
  \\Rightarrow R_n \leftarrow 3R_n+1
\end{verbatim}
3. A partial function $f$ is called *extendible* if there exists a total recursive $g$ that extends $f$, i.e. whenever $x \in \text{dom}(f)$, then $f(x) = g(x)$. Show that there exist partial recursive functions that are not extendible. Hint: consider $\phi(x) + 1$.

Suppose $\phi_e$ extends $\phi(x) + 1$, i.e., $\phi_e(x) = \phi(x) + 1$.

Consider $\phi_e(e)$. Since $\phi_e$ is total, $\phi(e)$, and

$$\phi_e(e) = \phi_e(e) + 1$$

Hence no such $\phi_e$ exists.
4. For a (possibly partial) function \( f : A \to B \) and \( U \subseteq A \), write \( f[U] = \{ f(x) | x \in U, f(x) \downarrow \} \).

(a) Prove or give a counterexample: if \( f : \mathbb{N} \to \mathbb{N} \) is recursive and \( U \) is r.e., then \( f[U] \) is r.e.

(b) Prove or give a counterexample: if \( f : \mathbb{N} \to \mathbb{N} \) is total recursive and \( U \) is recursive, then \( f[U] \) is recursive.

\[ a) \quad \text{True. We have} \]
\[ x \in f[A] \iff \exists y. (y \in A \land f(y) = x) \]

The set \( \{ (x, y) | y \in A, f(x) = y \} \) is r.e. since \( f \) is recursive and \( A \) is r.e.

Projections of r.e. sets are r.e. again, so \( f[A] \) is r.e.

\[ b) \quad \text{False. Every r.e. set is the image of some total recursive function.} \]
So take \( f \) tot. rec. with \( f[\mathbb{N}] = K \).
5. If \( f \) is a total function, then we write \( \text{Fix}(f) = \{ e | \phi_e = \phi_{f(e)} \} \) for the set of fixed points of \( f \).

(a) Explain why for any total recursive \( f \), the set \( \text{Fix}(f) \) is non-empty.

(b) Prove that for any total recursive \( f \), the set \( \text{Fix}(f) \) is in fact infinite. Hint: suppose \( \text{Fix}(f) \) were finite. Then let \( e \) be a code such that \( \phi_e \) is different from \( \phi_i \) for all \( i \in \text{Fix}(f) \). Then define \( g \) by

\[
g(x) \cdot y = \begin{cases} 
\phi_e(y) & \text{if } x \in \text{Fix}(f) \\
\phi_i(y) & \text{otherwise.}
\end{cases}
\]

Explain why \( g \) is total recursive, and prove that \( g \) does not have a fixed point, contradicting the fixed point theorem.

(a) This is simply the fixed point theorem

(b) Suppose \( x \in \text{Fix}(g) \), i.e., \( \phi_x = \phi_{g(x)} \), i.e.,

\[
x \cdot y = g(x) \cdot y \quad \text{for all } y.
\]

Now there are two cases:

Case 1. \( x \in \text{Fix}(f) \). Then \( x \cdot y = g(x) \cdot y = e \cdot y \), hence \( \phi_x = \phi_e \). But \( e \) was chosen so that \( \phi_e \neq \phi_i \) for all \( i \in \text{Fix}(f) \). Contradiction.

Case 2. \( x \notin \text{Fix}(f) \). Then \( x \cdot y = g(x) \cdot y = f(x) \cdot y \), meaning \( x \in \text{Fix}(f) \). Contradiction.

Conclusion: \( g \) doesn't have a fixed point.

But \( g \) is recursive since \( \text{Fix}(f) \) is finite...
6. In this question we write $T(e, x, y)$ for the T-predicate, which states that $y$ is the code of a computation of the function with code $e$ on input $x$; we also write $W_e$ for the domain of the function with code $e$. We define:

$$W_{e,s} = \{x | \exists y \leq s.T(e, x, y)\}.$$

(a) Show that for each $e, s$ the set $W_{e,s}$ is finite.

(b) Prove that $W_e = \bigcup_{s \in \mathbb{N}} W_{e,s}$.

\[\text{a) For fixed } e, s \text{ there are only finitely}
\]

\[\text{many } y \leq s. \text{ Each such } y \text{ with } T(e, x, y)
\]

\[\text{codes a successful computation of } y_e \text{ on an}
\]

\[\text{input } x. \text{ Since } T(e, x, y) \land T(e, x', y) \rightarrow x = x',
\]

\[\text{the set } W_{e,s} \text{ is finite.}
\]

\[\text{b) First, } W_{e,s} \subseteq W_e \text{ for all } s:
\]

\[\text{If } T(e, x, y) \text{ for some } y \leq s, \text{ then } x \in W_e.
\]

\[\text{Then also } \bigcup_{s \in \mathbb{N}} W_{e,s} \subseteq W_e.
\]

\[\text{Conversely if } x \in W_e, \text{ there is a code } y
\]

\[\text{with } T(e, x, y). \text{ Take } s = y. \text{ Then}
\]

\[x \in W_{e,s} \leq y \leq s. \{x | \exists y \leq s. T(e, x, y)\}.
\]
7. Consider the universal partial application function \( \bullet : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N} \).

(a) Show that \( \bullet \) is not associative, not commutative, and that there is no two-sided unit (in the sense that there is no \( i \) with \( e \bullet i = e = i \bullet e \) for all \( e \)).

(b) Show that there exists a number \( j \) with the property that for all \( x, y \in \mathbb{N} \), we have
\[
(j \bullet x) \bullet y = (x \bullet y) \bullet (y \bullet x).
\]

\[\begin{align*}
\text{a) Let } a \text{ be a code for } f(x) = 0 \quad \text{(can choose } a > 0) \\
&\text{let } b \text{ be a code for } f(x) = a \\
&\text{Then: } (b \cdot a) \cdot x = a \cdot x = 0 \\
&\quad \text{let } b \cdot (a \cdot x) = b \cdot 0 = a \\
\text{Also: } a \cdot b = 0 \quad b \cdot a = a
\end{align*}\]

There is no unit: if \( e \) were a unit, let \( i \) be a code for the successor function. Then \( i \cdot e = e + 1 = i \cdot e \) but we can take another code \( j \) for successor, so that \( j \cdot e = e + 1 = j \). Contradiction.

\[\begin{align*}
\text{b) The function } f(x, y) = (x \cdot y) \cdot (y, x) \text{ is partial recursive. The result now is immediate from the } \text{Sin}_n \text{- theorem.}
\end{align*}\]