1. (a) Show that \( f(x, y) = x + y \) and \( g(x, y) = x \cdot y \) are primitive recursive functions.

If \( g_1, g_2, \ldots, g_n \in \text{PRIM}, \) and if \( R_1, R_2, \ldots, R_n \) are primitive recursive relations such that for each \( x \in \mathbb{N} \) exactly one of \( R_i(x) \) holds, \( 1 \leq i \leq n, \) show that \( f \) is primitive recursive where

\[
  f(x) = \begin{cases} 
    g_1(x) & \text{if } R_1(x) \text{ holds,} \\
    g_2(x) & \text{if } R_2(x) \text{ holds,} \\
    \vdots \\
    g_n(x) & \text{if } R_n(x) \text{ holds.} 
  \end{cases}
\]

(b) Write a Turing program for

\[
  h(x, y) = x \div y,
\]

and briefly describe why your program works.

(c) Let \( \{\varphi_x\}_{x \in \mathbb{N}} \) be a standard list of all the partial computable functions.

If the total function \( f \) is defined by

\[
  f(x) = \begin{cases} 
    \varphi_x(x) + 1 & \text{if } \varphi_x \text{ is total,} \\
    0 & \text{otherwise,}
  \end{cases}
\]

explain why \( f \) cannot be computable.

Deduce that if

\[
  \text{Tot} = \{ x \in \mathbb{N} \mid \varphi_x \text{ is total}\},
\]

then \( \text{Tot} \) is not computable.
2. (a) We define: $A$ is \textit{computably enumerable (c.e.)} if and only if $A = \emptyset$, or $A$ is the range of some computable function.

Show that: (i) If $A \subseteq \mathbb{N}$ is computable, then $A$ is computably enumerable, and (ii) Every c.e. set is the halting set $W_i$ of some Turing machine.

[You may assume that every partial computable function is Turing computable.]

(b) Show that the following sets are c.e:

(i) $K_1 = \{x \mid W_x \neq \emptyset\}$, where you can assume that, for each $x \in \mathbb{N}$, $W_x$ is the $x^{\text{th}}$ computably enumerable set in some standard listing, and

(ii) $K = \{x \mid x \in W_x\}$.

Show that there exists a computably enumerable set which is not computable.

[You should carefully state any basic results of computability theory which you use.]

(c) Show that there exists a Turing machine $T$ with an unsolvable halting problem.

Deduce that the halting problem for the \textit{Universal Turing Machine} is unsolvable.

3. (a) We say $A \subseteq \mathbb{N}$ is \textit{creative} if and only if

1) $A$ is c.e., and

2) There is a computable function $f$ such that for each $e$

$$W_e \subseteq \overline{A} \Rightarrow f(e) \in \overline{A} - W_e,$$

where $\{W_e\}_{e \in \mathbb{N}}$ is a standard list of all c.e. sets.

Show that if $C$ is a creative set then

(i) $\overline{C} \neq \emptyset$,

(ii) For each $n \in \mathbb{N}$, if there exist $n$ members of $\overline{C}$ then there exist $n+1$ such members,

(iii) $\overline{C}$ contains an infinite c.e. subset.

[You may assume that for any finite set $X$ we can effectively find an $i$ such that $X = W_i$.]

(b) (i) Prove the \textit{Fixed Point Theorem} for a computable function $f$.

(ii) Explain why there is a computable function $f$ such that $W_{f(n)} = \{n\}$ for every $n \in \mathbb{N}$.

Say also why it is that every c.e. set $A$ has infinitely many distinct indices $e$ with $A = W_e$.

(iii) We say that a set $A$ is an \textit{index set} if for all $x, y \in \mathbb{N}$ we have

$$[x \in A \& W_x = W_y] \implies y \in A.$$

Let $K = \{x \mid x \in W_x\}$. By using the fixed point theorem, or otherwise, show that $K$ is not an index set.

**continued ...**
4. (a) Define the notions $A \leq_T B$ (that is, $A$ is Turing reducible to $B$), and $A \equiv_T B$ (that is, $A$ is Turing equivalent to $B$), where $A, B \subseteq \mathbb{N}$.

   (i) Show that $\equiv_T$ is an equivalence relation over the sets of natural numbers.

   (ii) Show that $\leq$ (the ordering induced by $\leq_T$ on the equivalence classes under $\equiv_T$) is a partial ordering on $\mathcal{D}$ (the set of all Turing degrees).

(b) The Turing jump $B'$ of $B \subseteq \mathbb{N}$ is defined to be

$$B' = \{\langle m, n \rangle \mid m \in W^B_n\},$$

where $\{W^B_n\}_{n \in \mathbb{N}}$ is a standard list of all $B$-c.e. sets.

   (i) Show that $B'$ is c.e. in $B$, and that if $X$ is c.e. in $B$ then $X \leq_m B'$.

   (ii) Show that if $B \leq_T A$ and $X$ is computably enumerable in $B$, then $X$ is computably enumerable in $A$.

   Deduce that if $A \equiv_T B$ then $A' \equiv_T B'$.

5. Either:

   (a) Show that there exists a pair of incomparable Turing degrees below $0'$.

   Or:

   (b) Outline briefly a proof of the following extension of the Friedberg-Muchnik Theorem:

   There exists an infinite sequence $\{a_i \mid i \geq 0\}$ of computably enumerable Turing degrees, such that for each $i \neq j$ we have $a_i \not\leq a_j$.

6. Write an essay, covering not more than three pages, describing the background to, and consequences of, Alan Turing’s discovery of the existence of a Universal Turing Machine.

   Your answer should contain enough mathematical content to show a good grasp of the notions and results involved, and enough discussion of these to show an understanding of the broader context.

END