Mr. T. Naughton.

Attempt any THREE questions. Time Allowed: 2 hours.

1. (a) **The table of behaviour of a TM to accept** $L$. The start state is 00. The accept state is 99. [8 marks]

<table>
<thead>
<tr>
<th>$S_i$</th>
<th>$R$</th>
<th>$S_f$</th>
<th>$W$</th>
<th>$M$</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>$a$</td>
<td>01</td>
<td>−</td>
<td>R</td>
</tr>
<tr>
<td>01</td>
<td>$a$</td>
<td>01</td>
<td>$a$</td>
<td>R</td>
</tr>
<tr>
<td>01</td>
<td>$b$</td>
<td>02</td>
<td>$b$</td>
<td>R</td>
</tr>
<tr>
<td>02</td>
<td>$b$</td>
<td>02</td>
<td>$b$</td>
<td>R</td>
</tr>
<tr>
<td>02</td>
<td>−</td>
<td>03</td>
<td>−</td>
<td>L</td>
</tr>
<tr>
<td>03</td>
<td>$b$</td>
<td>04</td>
<td>−</td>
<td>L</td>
</tr>
<tr>
<td>04</td>
<td>$b$</td>
<td>05</td>
<td>−</td>
<td>L</td>
</tr>
<tr>
<td>05</td>
<td>$b$</td>
<td>05</td>
<td>$b$</td>
<td>L</td>
</tr>
<tr>
<td>05</td>
<td>−</td>
<td>05</td>
<td>$a$</td>
<td>L</td>
</tr>
<tr>
<td>05</td>
<td>−</td>
<td>00</td>
<td>−</td>
<td>R</td>
</tr>
<tr>
<td>00</td>
<td>−</td>
<td>99</td>
<td>−</td>
<td>R</td>
</tr>
</tbody>
</table>

(b) Illustration of how a reduction can be used to prove nonmembership of **a class**. Given a problem $x$ not a member of class $A$, by finding a reduction $x \leq y$ you would prove that $y$ is not a member of $A$ either. [5 marks]
(c) **Proof that $W$ is decidable.**

Let $X = \text{“On input } \langle M, w \rangle :$
1. Let $a = 2^{\mid w \mid}$.
2. Run $M$ on $w$ and count the number of timesteps.
3. If $M$ halts before $a$ timesteps, reject, otherwise accept.”

$X$ is a TM that decides $W$, therefore $W$ is decidable.

(d) **Proof that PRINTERPROBLEM $\in NP$.** The certificate $c = (Q_1, Q_2)$ is the two lists of jobs for the two printers.

Let $V = \text{“On input } (n, P, t, (Q_1, Q_2)) :$
1. Check that each element of $Q_1 \cup Q_2$ is in $P$ : $n \times n = O(n^2)$.
2. Check that each element of $P$ is in $Q_1 \cup Q_2$ : $n \times n = O(n^2)$.
3. Check that $Q_1 \cap Q_2 = \emptyset$ : $n \times n = O(n^2)$.
4. Check that the sum of $Q_1 \leq t$ and that the sum of $Q_2 \leq t : 2n = O(n)$.
5. If all checks are passed, accept : $1 = O(1)$.

Machine $V$ verifies PRINTERPROBLEM. $V$ requires $O(n^2)$ timesteps in total, so PRINTERPROBLEM is in $NP$.

2. (a) **Definition of a model of computation.** A model of computation is a list of assumptions about the capabilities of a computing device.

(b) **Proof that INFINITE$_{\text{TM}}$ is undecidable.**

i. AT$_{\text{TM}} \leq$ INFINITE$_{\text{TM}}$
ii. INFINITE$_{\text{TM}}$
iii. AT$_{\text{TM}}$
iv. INFINITE$_{\text{TM}}$
v. $\langle M, w \rangle$
vi. “On input $x :$
1. If $x \in \{01, 11, 100\}$, then accept $x$. 
2. Run $M$ on $w$.
3. If $M$ accepts $w$, then accept $x$.”

vii. $\langle M' \rangle$
viii. INFINITE$_{\text{TM}}$
ix. AT$_{\text{TM}}$
x. AT$_{\text{TM}}$
3. (a) **Proof that NEVEROVERFLOW\(_J\) is undecidable.** [15 marks]
   i. \(A_j \leq \text{NEVEROVERFLOW}_J\)
   ii. \(\text{NEVEROVERFLOW}_J\)
   iii. \(A_j\)
   iv. \(\text{NEVEROVERFLOW}_J\)
   v. \(\langle J, w \rangle\)
   vi. “class Mprime {
       public static void main(String args[]) {
           int a = 0;
           if \((J(w) == \text{accept})\) {
               while \((1 == 1)\) {
                   a++;
               }
           }
       }
   }
   ”
   vii. \(\langle M', a \rangle\)
   viii. \(\text{NEVEROVERFLOW}_J\)
   ix. \(A_j\)
   x. \(A_j\)

(b) **Proof that NEVEROVERFLOW\(_J\) is not Turing recognisable.** We construct a TM \(M\) to recognise NEVEROVERFLOW\(_J\) as follows.

   \(X = \text{“On input} \langle J, v \rangle \text{;}\)
   1. Run \(J\) checking the value in \(v\) at each timestep.
   2. If \(v\) overflows, accept.”

   \(X\) recognises \(\text{NEVEROVERFLOW}_J\) therefore \(\text{NEVEROVERFLOW}_J\) is Turing recognisable. Since \(\text{NEVEROVERFLOW}_J\) is undecidable, and \(\text{NEVEROVERFLOW}_J\) is Turing recognisable, this proves that \(\text{NEVEROVERFLOW}_J\) is not Turing recognisable.

(c) i. \(\text{NEVEROVERFLOW}_{TM} = \{\langle J, v \rangle : J \text{ is a Java program, } v \text{ is an integer variable declared in } J, \text{ and when } J \text{ is run variable } v \text{ overflows at least once}\}\) [2 marks]
   ii. **Proof that \(\text{NEVEROVERFLOW}_{TM}\) is Turing recognisable.** This has been proved in 3b above. [3 marks]

4. (a) **Definition of the Church-Turing thesis.** Turing machines are equivalent to all other reasonable computing devices. [5 marks]
   (b) **Proof that \(2^\Sigma^*\) is uncountable.** Assume that \(2^\Sigma^*\) is countable. Then it should be possible to create a list (infinite in this case) containing all of the elements of \(2^\Sigma^*\) (all of the languages over \(\Sigma\)). Consider such a list of languages, and represent each language by an infinite sequence over \(\{T, F\}\) where a \(T\) at the \(n\)th position indicates that the \(n\)th word in the lexicographic ordering of \(\Sigma^*\) is in that language, and a \(F\) at the \(n\)th position indicates that the \(n\)th word in the
lexicographic ordering of $\Sigma^*$ is not in that language. We can represent this infinite list of infinite sequences as a table, infinite in both directions. Now, if we extract the diagonal of this table, and convert each $T$ to $F$ and each $F$ to $T$, we get a valid representation of a language over $\Sigma$ that is not in the list. A contradiction, because this list was supposed to contain all such languages. Therefore our assumption was wrong and $2^\Sigma^*$ must be uncountable.

(c) **Placement of each language and its complement in the space of languages.** [10 marks]
The solutions will be given in the following form (smallest class the language is in, smallest class its complement is in).

i. (T-r, $2^\Sigma^*$)
ii. (EXP, EXP)
iii. (NP, coP)
iv. ($2^\Sigma^*$, $2^\Sigma^*$)
v. (P, P)