OLLSCOIL NA hÉIREANN, MÁ NUAD

NATIONAL UNIVERSITY OF IRELAND, MAYNOOTH

THIRD COMPUTER SCIENCE EXAMINATION

FOURTH COMPUTER SCIENCE EXAMINATION

THIRD COMPUTER SCIENCE AND SOFTWARE ENGINEERING EXAMINATION

FOURTH COMPUTER SCIENCE AND SOFTWARE ENGINEERING EXAMINATION

MASTERS OF COMPUTER SCIENCE (YEAR 1) EXAMINATION

MASTERS OF COMPUTER SCIENCE (YEAR 2) EXAMINATION

LAB TEST 2

PAPER CS355/SE307/CS403

COMPUTATION AND COMPLEXITY

T. Naughton.

Attempt ALL questions. Time Allowed: 35 minutes.

1. **Proof that $\text{VARPOSITIVE}_\text{J}$ is undecidable.** We will use a mapping reduction to prove the reduction $\text{A}_\text{J} \leq \text{VARPOSITIVE}_\text{J}$. Assume that $\text{VARPOSITIVE}_\text{J}$ is decidable. The function f that maps instances of A_J to instances of $\text{VARPOSITIVE}_\text{J}$ is performed by TM F given by the following pseudocode. [20 marks]

   $F =$ "On input $\langle J, w \rangle$:
   1. Construct the following M' given by the following pseudocode.
      $M' =$ "On any input :
         int $a = -1$;
         Run J on w.
         If J accepts, assign $a = 5$."
   2. Output $\langle M', a \rangle$"

   Now, $\langle M', a \rangle$ is an element of $\text{VARPOSITIVE}_\text{J}$ iff $\langle J, w \rangle$ is an element of A_J. So using f and the assumption that $\text{VARPOSITIVE}_\text{J}$ is decidable, we can decide A_J. A contradiction. Therefore, $\text{VARPOSITIVE}_\text{J}$ is undecidable.

2. **Proof that SQUAREPAIRS can be verified in polynomial time.** We will choose as certificate c an array of length two containing the indices of x and y in A (let $c[0]$ be the index for x and let $c[1]$ be the index for y). [15 marks]

   M = "On input $\langle A, c\rangle$:
   1. If $A[c[0]] \times A[c[0]] = A[c[1]]$, then accept; otherwise reject : $3 = O(1)$"

   Machine M verifies SQUAREPAIRS. It requires $O(1)$ timesteps (which is polynomial in the length of A). Therefore, SQUAREPAIRS can be verified in polynomial time.

   Alternative solution to this problem:

   Proof that SQUAREPAIRS can be verified in polynomial time. We will choose as certificate c the index of x in A.

   M = "On input $\langle A, c\rangle$:
   1. For each element i in A, check if $A[c] \times A[c] = A[i]$: $3n = O(n)$
   2. If any suitable i is found, accept; otherwise reject : $1 = O(1)$"

   Machine M verifies SQUAREPAIRS. It requires $O(n)$ timesteps in total. Therefore, SQUAREPAIRS can be verified in polynomial time.