1. Derive the number of different ways that a $d$-dimensional hypercube can be labeled. (5 marks)

2. The distance between nodes $u$ and $v$ in a hypercube is the length of the shortest path from $u$ to $v$. Given a $d$-dimensional hypercube and a designated source node $s$, how many nodes are of distance $i$ from the node $s$ where $0 \leq i \leq d$? Derive your answer. (5 marks)

3. The dual of all-to-all broadcast is all-to-all reduction, in which each node is the destination of an all-to-one reduction. For example, consider the scenario where $p$ nodes have a vector of $p$ elements each, and the $i$th node (for all $i$ such that $0 \leq i < p$) gets the sum of the $i$th elements of all the vectors. Propose a good parallel algorithm to perform all-to-all reduction on a hypercube with addition as the associative operator. If each message contains $m$ words and $t_{\text{add}}$ is the time to perform one addition, how much time does your algorithm take (in terms of $m$, $p$, $t_{\text{add}}$, $t$, and $t_c$)? (20 marks)

*Hint:* In all-to-all broadcast, each node starts with a single message and collects $p$ such messages by the end of the operation. In all-to-all reduction, each node starts with $p$ distinct messages but ends up with a single message.

4. The string matching problem is to find all occurrences of a particular substring called the pattern, in another string called the text. Propose a good parallel algorithm to solve the string matching problem with the use of split, send, receive and gather functions, and use diagrams in your illustrations. MPI program is not needed. You have to assume the existence of multiple matches, i.e., the substring can appear more than one time in the text. (20 marks)

5. Write a master/slave-style MPI program (you need to edit it, compile it, debug it and run it before you hand in the source file) that finds the smallest positive root of the equation $f(x) = -2 + \sin x + \sin x^2 + \sin x^3 + ... + \sin x^{1000}$. The root is the unique value $r$ between 0 and 1 such that $f(r) = 0$. The program should divide the interval $[0, 1]$ into several sub-intervals and create a set of tasks, one for each sub-interval where the function changes from negative to positive. The algorithm can then iterate, dividing this sub-interval into pieces for parallel execution again. When the subinterval size becomes less than $10^{-11}$, the master should terminate and print the root of the function. (50 marks)

Hand in your source codes named as root.c for questions 5 by an email attachment to scitaysc@nus.edu.sg. Comment your name on the first line of source file. Also print the source codes in hardcopy, and hand it in together with the answers to Questions 1 to 4.

(This individual assignment carries 15% of the final score. Please hand in your work on time. We practice the honour system.)