CSI 5165 Combinatorial Algorithms — Fall 2003
Computer Science — University of Ottawa

Homework Assignment #2 (100 points, weight 15%)
Due: Tuesday, November 4, at 9:00 a.m. (in lecture)

---

1. (30 points) **Backtracking Algorithms for Graph Colouring**
   Define choice sets and describe backtracking algorithms for the following problems:
   - Find all k-vertex colourings of a graph G.
   - Find a minimum colouring of a graph G, that is, a k-vertex colouring for the smallest k possible. Use bounding for your algorithm; note however, that this is a minimization problem, so you would need to find a lower bound.

2. (35 points) **Backtracking program for non-linear codes**
   If $\{x, y\} \in \{0,1\}^n$, then recall that $\textsc{dist}(x, y)$ denotes the Hamming distance between x and y. A *non-linear code* of length n and minimum distance d is a subset $\mathcal{C} \subseteq \{0,1\}^n$ such that $\textsc{dist}(x, y) \geq d$ for all $x, y \in \mathcal{C}$. Denote by $A(n, d)$ the maximum number of n-tuples in a length n non-linear code of minimum distance d.
   - Describe a backtracking algorithm to compute $A(n, d)$.
   - Implement your backtracking algorithm and compute $A(n, 4)$, for $n \leq 8$.
   - Bonus challenge (10 marks): determine the values of $A(9, 4)$ and $A(10, 4)$; these values are more difficult to obtain and are 20 and 40, respectively. Only work on the bonus challenge after all other parts of your assignent have been completed. The value for $A(11, 4)$ is not known, but it is known to be between 72 and 79.

3. (35 points) **Hill Climbing to Find Transversal Triple Systems**
   A transversal triple system TTS(n) is a set system $(X, \mathcal{B})$, such that:
   (a) $X = X_1 \cup X_2 \cup X_3$, with $|X| = 3n$ and $|X_i| = n$, for $i = 1, 2, 3$.
   (b) For each $B \in \mathcal{B}$ we have that $|B| = 3$ and $|B \cap X_i| = 1$.
   (c) For every $x \in X_i$ and $y \in X_j$ with $i \neq j$, there exists a unique block $B \in \mathcal{B}$ such that $\{x, y\} \subseteq B$.

   Develop a hill-climbing algorithm to construct a transversal triple system TTS(n). (Hint: design a heuristic similar to that used in the algorithm for constructing Steiner Triple Systems). Show your pseudo-code and an implementation program. Test your algorithm for various values of n (repeat 5 times for each n). Report on the total number of solutions visited as well on the total running time.

Note: For all problems, clarity and efficiency will be taken into account when marking.