CHENNAI MATHEMATICAL INSTITUTE

M.Sc. / Ph.D. Programme in Computer Science

Entrance Examination, 25 May 2012

This question paper has 5 printed sides. Part A has 10 questions of 3 marks each. Part B has 7 questions of 10 marks each. The total marks are 100.

Part A

1. Let $L \subseteq \{0,1\}^*$. Which of the following is true?

   (a) If L is regular, all subsets of L are regular.

   (b) If all proper subsets of L are regular, then L is regular.

   (c) If all finite subsets of L are regular, then L is regular.

   (d) If a proper subset of L is not regular, then L is not regular.

2. Let T be a tree on 100 vertices. Let n_i be the number of vertices in T which have exactly i neighbors. Let $s = \sum_{i=1}^{100} i \cdot n_i$. Which of the following is true?

   (a) $s = 99$

   (b) $s = 198$

   (c) $99 < s < 198$

   (d) None of the above

The next two questions are based on the following program. In the program, we assume that integer division returns only the quotient. For example $7/3$ returns 2 since 2 is the quotient and 1 is the remainder.

```
mystery(a,b){
   if (b == 0) return a;
   if (a < b) return mystery(b,a);
   if (eo(a,b) == 0){
      return(2*mystery(a/2,b/2));
  }
  if (eo(a,b) == 1){
         return(mystery(a,b/2));
  }
  if (eo(a,b) == 2){
                 return(mystery(a/2,b));
  }
  if (eo(a,b) == 3){
         return (mystery((a-b)/2,b));
  }
}
eo(a,b){
```

```
  if ((a/2)*2 == a and (b/2)*2 == b) return 0; end;
  if ((a/2)*2 < a  and (b/2)*2 == b) return 1; end;
  if ((a/2)*2 == a  and (b/2)*2 < b) return 2; end;
  return 3;
}
```

3. `mystery(75,210)` returns

   (a) 2 (b) 6 (c) 10 (d) 15

4. When a and b are n bit positive numbers the number of recursive calls to `mystery` on input a, b is

   (a) $O(n)$ (b) $O(\log\log n)$ (c) $O(\log n)$ (d) $O(n^{1/2})$

5. Amma baked a cake and left it on the table to cool. Before going for her bath, she told her four children that they should not touch the cake as it was to be cut only the next day. However when she got back from her bath she found that someone had eaten a large piece of the cake. Since only her four children were present at home when this happened, she questioned them about who ate a piece of the cake. The four answers she got were:

   Lakshmi: Aruna ate the piece of cake.
   Ram: I did not eat the piece of cake.
   Aruna: Varun ate the cake.
   Varun: Aruna lied when she said I had eaten the piece of cake.

   If exactly one of them was telling the truth and exactly one of them actually ate the piece of cake, who is the culprit that Amma is going to punish?

   (a) Lakshmi (b) Ram (c) Aruna (d) Varun

6. A basket of fruit is being arranged out of apples, bananas, and oranges. What is the smallest number of pieces of fruit that should be put in the basket in order to guarantee that either there are at least 8 apples or at least 6 bananas or at least 9 oranges?

   (a) 9 (b) 10 (c) 20 (d) 21

7. A man has three cats. At least one is male. What is the probability that all three are male?

   (a) $\frac{1}{2}$ (b) $\frac{1}{7}$ (c) $\frac{1}{8}$ (d) $\frac{3}{8}$

8. You are given two sorting algorithms A and B that work in time $O(n \log n)$ and $O(n^2)$, respectively. Consider the following statements:

   (I) Algorithm A will sort any array faster than algorithm B.

   (II) On an average, algorithm A will sort a given array faster than algorithm B.

   (III) If we need to implement one of the two as the default sorting algorithm in a system, algorithm A will always be preferable to algorithm B.

Which of the statements above are true?

(a) I, II and III (b) I and III (c) II and III (d) None of them

9. Consider the following programming errors:

   (I) Type mismatch in an expression.

   (II) Array index out of bounds.

   (III) Use of an uninitialized variable in an expression.

   Which of these errors will typically be caught at compile-time by a modern compiler.

   (a) I, II and III (b) I and II (c) I and III (d) None of them

10. Consider the following functions `f` and `g`.

```
f(){                    g(){
    x = x-50;               a = a+x;
    y = y+50;               a = a+y;
}                       }
```

   Suppose we start with initial values of `100` for `x`, `200` for `y`, and `0` for `a`, and then execute `f` and `g` in parallel—that is, at each step we either execute one statement from `f` or one statement from `g`. Which of the following is *not* a possible final value of `a`?

   (a) 300 (b) 250 (c) 350 (d) 200

Part B

1. Let $G = (V, E)$ be a graph where $|V| = n$ and the degree of each vertex is strictly greater than $n/2$. Prove that G has a Hamiltonian path. (Hint: Consider a path of maximum length in G.)

2. For a binary string $x = a_0 a_1 \cdots a_{n-1}$ define $val(x)$ to be

$$\sum_{0 \leq i < n} 2^{n-1-i} \cdot a_i.$$

   Let $\Sigma = \{(0,0), (0,1), (1,0), (1,1)\}$.

   (i) Construct a finite automaton that accepts the set of all strings

$$(a_0, b_0)(a_1, b_1) \cdots (a_{n-1}, b_{n-1}) \in \Sigma^*$$

   such that

$$val(b_0 b_1 \cdots b_{n-1}) = 2 \cdot val(a_0 a_1 \cdots a_{n-1}).$$

(ii) Construct a finite automaton that accepts exactly those strings

$$(a_0, b_0)(a_1, b_1) \cdots (a_{n-1}, b_{n-1}) \in \Sigma^*$$

such that

$$val(b_0 b_1 \cdots b_{n-1}) = 4 \cdot val(a_0 a_1 \cdots a_{n-1}).$$

3. Let A be array of n integers, sorted so that $A[1] \leq A[2] \leq ..A[n]$. Suppose you are given a number x and you wish to find out if there there indices k, l such that $A[k]+A[l] = x$.

   (i) Design an $O(n \log n)$ time algorithm for this problem.

   (ii) Design an $O(n)$ algorithm for this problem.

4. You have an array `A` with n objects, some of which are identical. You can check if two objects are equal but you cannot compare them in any other way—i.e. you can check `A[i] == A[j]` and `A[i] != A[j]`, but comparisons such as `A[i] < A[j]` are not meaningful. (Imagine that the objects are JPEG images.)

   The array `A` is said to have a majority element if strictly more than half of its elements are equal to each other. Use divide and conquer to come up with an $O(n \log n)$ algorithm to determine if `A` has a majority element.

5. Given an undirected weighted graph $G = (V, E)$ with non-negative edge weights, we can compute a minimum cost spanning tree $T = (V, E')$. We can also compute, for a given source vertex $s \in V$, the shortest paths from s to every other vertex in V.

   We now increase the weight of every edge in the graph by 1. Are the following true or false, regardless of the structure of G? Give a mathematically sound argument if you claim the statement is true or a counterexample if the statement is false.

   (i) T is still a minimum cost spanning tree of G.

   (ii) All the shortest paths from s to the other vertices are unchanged.

6. A certain string-processing language offers a primitive operation which splits a string into two pieces. Since this operation involves copying the original string, it takes n units of time for a string of length n, regardless of the location of the cut.

   Suppose, now, that you want to break a string into many pieces. The order in which the breaks are made can affect the total running time. For example, if you want to cut a 20-character string at positions 3 and 10, then making the first cut at position 3 incurs a total cost of $20 + 17 = 37$, while doing position 10 first has a better cost of $20 + 10 = 30$.

   Give a dynamic programming algorithm that, given the locations of m cuts in a string of length n, finds the minimum cost of breaking the string into $m + 1$ pieces. You may assume that all m locations are in the interior of the string so each split is non-trivial.

7. We use the notation `[x1,x2,...,xn]` to denote a list of integers. `[]` denotes the empty list, and `[n]` is the list consisting of one integer `n`. For a nonempty list `l`, `head(l)` returns the first element of `l`, and `tail(l)` returns the list obtained by removing the first element from `l`. The function `length(l)` returns the length of a list. For example,

- `head([11,-1,5]) = 11`, `tail([11,-1,5]) = [-1,5]`.
- `head([7]) = 7`, `tail([7]) = []`.
- `length([]) = 0`, `length([7]) = 1`, `length([11,-1,5]) = 3`.

We use `or`, `and` and `not` to denote the usual operations on boolean values `true` and `false`.

Consider the following functions, each of which takes a list of integers as input and returns a boolean value.

```
f(l)
   if (length(l) < 2) then return(true)
   else return(g(l) or h(l))

g(l)
   if (length(l) < 2) then return(true)
   else
      if (head(l) < head(tail(l)) then return h(tail(l))
      else return(false)

h(l)
   if (length(l) < 2) then  return(true)
   else
      if (head(l) > head(tail(l)) then return g(tail(l))
      else return(false)
```

When does `f(l)` return the value `true` for an input `l`? Explain your answer.