
QUALIFYING EXAMINATION

THEORETICAL COMPUTER SCIENCE

PART I: ALGORITHMS

MONDAY, FEBRUARY 22, 2010

Problem 1

Let $G = (V, E)$ be an undirected bipartite graph with bipartition A, B .

- Prove that G has k edge-disjoint perfect matchings if it is k -regular.
- Give an example to show that having minimum degree k is not sufficient for the existence of k edge-disjoint perfect matchings.
- Let $|A| = |B| = n$. Show that G has k edge disjoint perfect matchings if and only if for each $A' \subseteq A$ and $B' \subseteq B$, there are at least $k(|A'| + |B'| - n)$ edges between A' and B' .

Problem 2

Consider the following random process that is done in rounds. It starts with m balls and n bins. In each round, each ball is placed independently into a bin chosen uniformly at random from the n bins. Any ball that lands in a bin by itself is discarded forever. The process continues with the remaining balls until there are no balls left. Show that the expected number of rounds for the process to terminate is $O(\log \log n)$ if we start with $m = n$.

Problem 3

Substring compression is a compression scheme reminiscent of Lempel-Ziv encoding. Let T be a string of length n over some fixed alphabet Σ . To compress T , we can replace the second (or later) occurrence of any substring of T with a pair of indices describing its first occurrence.

For example, in the string HOCUSPOCUS, we can replace the second occurrence of the repeated substring OCUS with the pair [2, 5] to get the compressed representation HOCUSP[2, 5]. More subtly, we can also compress overlapping occurrences of the same substring; for example, HA[1, 16] is a valid substring compression of the 18-character string HAHAAAAAAAAAAAAAAAAA.

The *length* of a compressed string is the number of raw characters plus the number of indices. For example, the compressed strings HOCUSP[2, 5] and HA[1, 16] have lengths 8 and 4, respectively. (The brackets and commas are syntactic sugar.)

Here is the algorithm to decompress a compressed string of length m :

```

RSDECOMPRESS( $C[1..m]$ ):
   $n \leftarrow 1$ 
   $i \leftarrow 1$ 
  while  $i \leq m$ 
    if  $C[i] \in \Sigma$             $\langle\langle \text{character} \rangle\rangle$ 
       $T[n] \leftarrow C[i]$ 
       $n \leftarrow n + 1$ 
       $i \leftarrow i + 1$ 
    else                        $\langle\langle \text{substring indices} \rangle\rangle$ 
      for  $j \leftarrow C[i]$  to  $C[i + 1]$ 
         $T[n] \leftarrow T[j]$ 
         $n \leftarrow n + 1$ 
       $i \leftarrow i + 2$ 
  return  $T[1..n]$ 

```

1. Describe and analyze an efficient algorithm to compute the minimum-cost substring compression of a given string.
2. Prove that every n -character string has a repeated substring compression of length $O(n/\log n)$. (The hidden constant may depend on the size of the alphabet Σ .)
3. Prove that for all n , there is an n -character string X_n such that *every* repeated substring compression of X_n has length $\Omega(n/\log n)$. (Again, the hidden constant may depend on the size of the alphabet Σ .)
4. Give a constructive proof for part (c). That is, for each integer n , describe a string $X_n \in \{0, 1\}^n$ such that *every* repeated substring compression of X_n has length $\Omega(n/\log n)$.

Problem 4

Consider a graph G with minimum degree δ , and girth $2k + 1$, where k is some integer. The *girth* of a graph is the minimum length cycle in this graph.

1. Prove that G has at least $(\delta - 1)^k$ vertices.
2. Prove that a graph with girth $\geq 2k + 1$ has at most $O(n^{1+1/k})$ edges.
3. A graph $H \subseteq G$ is a t -spanner for G , if for any two vertices $u, v \in V(G)$ we have that $d_G(u, v) \leq d_H(u, v) \leq t \cdot d_G(u, v)$, here $d_G(u, v)$ denotes the (unweighted) shortest path between u and v in G .

Consider the algorithm that constructs the spanner by starting from the empty graph $H = (V(G), \emptyset)$, and considering the edges of G in arbitrary order e_1, \dots, e_m . In the i th iteration, the algorithm computes the distance between the two endpoints of e_i in G and in the current graph H , and add e_i to H if $d_H(u, v) > t$.

It is not too hard to argue that G is a t -spanner for H . Prove that if $t > 2$ is odd then G has at most $O(n^{1+2/(t-1)})$ edges.

4. Construct a bipartite graph with $2n$ vertices such that it does not contain any cycle of length 4, and it contains at least (say) $n^{3/2}/8$ edges (but any constant instead of 8 is good enough). Prove that any such graph has at most $O(n^{3/2})$ edges.

(Hint: For the construction, use the probabilistic method. (There is also a direct algebraic construction.))