Algorithms and Complexity. Exercise session 1

Algorithm analysis

Order Compare the asymptotic order of growth of the following pairs of functions. In each case tell if $f(n) \in \Theta(g(n))$, $f(n) \in O(g(n))$ or $f(n) \in \Omega(g(n))$.

	$f(n)$	$g(n)$
a)	$100n + \log n$	$n + (\log n)^2$
b)	$\log n$	$\log n^2$
c)	$\frac{n^2}{\log n}$	$n(\log n)^2$
d)	$(\log n)^{\log n}$	$\frac{n}{\log n}$
e)	$\sqrt{n}$	$(\log n)^5$
f)	$n2^n$	3^n
g)	$2^{\sqrt{\log n}}$	$\sqrt{n}$

Division Analyze the schoolbook algorithm for division (stair division). Assume that each bit operation takes unit cost.

Euclid's algorithm Analyze Euclid's algorithm that finds the greatest common divisor between two integers. Do the analysis in terms of bit cost and unit cost. Below we present the algorithm and assume that $a \geq b$.

$gcd(a, b)=$
  if $b|a$ **then**
    $gcd \leftarrow b$
  else
    $gcd \leftarrow gcd(b, a \bmod b)$

Exponentiation with repeated squaring The following algorithm computes a power of 2 with the exponent which is itself a power of 2. Analyze the algorithm in terms of unit cost and bit cost.

Input: $m = 2^n$
Output: 2^m
$power(m) =$
  $pow \leftarrow 2$
  for $i \leftarrow 1$ **to** $\log m$ **do**
    $pow \leftarrow pow \cdot pow$
  return pow

Incidence matrix product In the lecture we showed how to represent a directed graph $G = \langle V, E\rangle$ by an incidence matrix of size $|V| \times |E|$. The element at position (i, j) is set to 1 iff $v_i \in e_j$, namely, the vertex v_i is covered by the edge e_j. Show what does the elements of matrix BB^T represent (B^T is the transposed matrix of B).

Bipartite graphs Describe and analyze an algorithm that determines whether a given graph is bipartite. The time complexity will be linear in the number of nodes and edges of the graph. A graph $G = (N, E)$ is bipartite iff the nodes can be divided into two disjoint sets U and V such that every edge connects a vertex in U to one in V.
