Digital Logic Systems

Recitation 3: Directed Graphs & Binary Representation

Guy Even   Moti Medina

School of Electrical Engineering Tel-Aviv Univ.

November 15, 2011

Directed Graphs: Some more notations

Recall that Theorem 4.6 stated that if $G = (V, E)$ is a rooted tree then there is a unique path from every vertex $v \in V$ to the root $r(G)$.
Hence, the following terms are well defined. Let $G = (V, E)$ denote a rooted tree.

- The depth of a vertex $v \in V$ is the length of the path from v to the root.
- The height of a rooted tree $G = (V, E)$ is maximum depth of a vertex in V.
- Note that a single isolated vertex is a rooted tree. Its height is zero.

Topological Sort of Rooted Trees

A topological sorting of a rooted tree $G = (V, E)$ can be computed as follows.

Topological Sort of a Rooted Tree $G = (V, E)$

Sort the vertices in each subtree hanging from the root, and order the root last.

Divisibility by Powers of 2

- Recall that the natural number represented by the binary string $A[n-1:0]$ is:

$$\langle A[n-1:0]\rangle \triangleq \sum_{i=0}^{n-1} A[i]\cdot 2^i.$$

- We already noticed that if the LSB of A is '0' then $\langle A[n-1:0]\rangle$ is even, furthermore, if both $A[0]$ and $A[1]$ equal '0' then the number $\langle A[n-1:0]\rangle$ is divisible by $4 = 2^2$, e.g. 100 represents the number 4, 1100 represents the number 12.
- We generalize this property in the following lemma.

Lemma (5.3)

Let $A[n-1:0]$ be a binary string, and let $a \triangleq \langle A[n-1:0]\rangle$, then a is divisible by 2^k if $A[i]=0$ for all $0 \leq i \leq k-1$.

Proof.

On the whiteboard. □

Word Length vs. Memory size in a PC

- Now we can discuss a matter that is close to our hearts, that is the relation between word length and memory size in a computer.
- A central processing unit (CPU) addresses its memory modules by a fixed size bit string which is called a word (nowadays it is typical that the CPU's word is 32 and 64 bits).

Word Length vs. Memory size in a PC: Units of measurement

- A byte (B) is 8 bit length word.
- A kilo-bit (Kb) is 2^{10} bits.
- Hence, a kilo-byte (KB) is $8 \cdot 2^{10}$ bits.
- A mega-bit (Mb) is $2^{10} \cdot 2^{10} = 2^{20}$ bits.
- Analogously, A mega-byte (MB) is $8 \cdot 2^{20}$ bits.
- A giga-bit (Gb) is $2^{10} \cdot 2^{20} = 2^{30}$ bits.
- Analogously, A giga-byte (GB) is $8 \cdot 2^{30}$ bits.
- If we consider words of ω bits, then a giga-word (GW) is $\omega \cdot 2^{30}$ bits.

Word Length vs.Memory size in a PC (cont.)

Question

What is the size of single memory module in GW if the CPU's word length is 32 bits?

Let $A[k-1:0]$ denote a k bit string. Let $a_k \triangleq \langle A[k-1:0] \rangle$. Since a_{32} attains values in $\{0, \ldots, 2^{32}-1\}$ it implies that the CPU can address 2^{32} different values, i.e. to address 2^{32} words. That means that if your personal computer (PC) has a 32-bit CPU, then there is no need to purchase more than 4 GW of memory.

Computing a Binary Representation

We consider algorithm $BR'(x, k)$ for computing a binary representation.

The algorithm's specification is as follows.

Inputs: $x \in \mathbb{N}$ and $k \in \mathbb{N}^+$, where x is a natural number for which a binary representation is sought, and k is the length of the binary string that the algorithm should output.

Output: The algorithm outputs "fail" or a k-bit binary string $A[k-1:0]$.

Functionality: The relation between the inputs and the output is as follows:

1. If $0 \le x < 2^k$, then the algorithm outputs a k-bit string $A[k-1:0]$ that satisfies $x = \langle A[k-1:0] \rangle$.
2. If $x \ge 2^k$, then the algorithm outputs "fail".

Computing a Binary Representation - Algorithm

The algorithm is as follows.

$BR'(x,k)$ - An LSB-to-MSB algorithm for computing a binary representation of a natural number a using k bits.

1. Base Cases:
   1. If $x \geq 2^k$ then return (fail).
   2. If $k = 1$ then return (x).
2. Reduction Rule:
   1. If x is even then return $(BR'(x/2, k-1) \circ 0)$.
   2. If x is odd then return $(BR'((x-1)/2, k-1) \circ 1)$.

Computing a Binary Representation - Correctness

The correctness of algorithm $BR'(x, k)$ is summarized in the following theorem.

Theorem

If $x \in \mathbb{N}$, $k \in \mathbb{N}^+$, and $x < 2^k$, then algorithm $BR'(x, k)$ returns a k-bit binary string $A[k-1:0]$ such that $\langle A[k-1:0] \rangle = x$.

Proof.

On the whiteboard. $\square$