CS 2800: Discrete Math Sept. 21, 2011

Complimentary Definitions

Lecturer: John Hopcroft *Scribe: June Andrews*

Number Theory

Greatest common divisor (GCD)

The *greatest common divisor* of two or more non-zero integers is the largest positive integer that divides each of the integers evenly (i.e., without a remainder). We use $\gcd(a_1, a_2, ..., a_n)$, where $n \geq 2$, to denote the greatest common divisor of non-zero integers $a_1, a_2, ..., a_n$. In general, $\gcd(a_1, a_2, ..., a_n)$ can be computed by finding the prime factorizations of $a_1, a_2, ..., a_n$. In the special case where $n = 2$, the GCD can also be computed by using the Euclidean algorithm.

Examples. The greatest common factor of 8 and 12 is 4, and the greatest common factor of $12, 20$, and 30 is 2. For any integer a, $\gcd(a, 0) = a$ and $\gcd(a, 1) = 1$.

Relatively Prime

Two integers a and b are said to be *coprime* or *relatively prime* if they have no common positive divisors other than 1. Equivalently, a and b are relatively prime if their greatest common divisor is 1. One writes $a \perp b$ to express that a and b are coprime.

Examples. The integers 25 and 14 are relatively prime because their GCD is 1, but 14 and 21 are not relatively prime because $7 > 1$ divides both of them. The integers 1 and -1 are coprime to every integer and are the only integers coprime to 0.

Proofs

Lemma

A *lemma* is a statement that is proved as an intermediate step to a larger result.

Example. Assignment 4 asked you to prove that if a and b are relatively prime and a divides bc, then a divides c. This statement, known as Euclid's lemma, is used in some proofs of the fundamental theorem of arithmetic.

Theorem

A *theorem* is a statement that has been proven using previously established results and previously accepted statements.

Corollary

A *corollary* is an immediate consequence of a statement that has already been proven. In other words, it is a statement that logically follows from a theorem with little or no proof.

Induction

Suppose that $\mathcal{S}_n$ is a statement involving a variable n and that we wish to prove that $\mathcal{S}_n$ is true for all natural numbers n. The *induction principle* states that we only need to prove the following two statements:

1. (Base Case) $\mathcal{S}_0$ is true.

2. (Inductive Step) For every $k \in \mathbb{N}$, if $\mathcal{S}_k$ is true, then $\mathcal{S}_{k+1}$ is true as well.

In other words, if statements (1) and (2) are true, then $\mathcal{S}_n$ is true for all $n \in \mathbb{N}$.

An informal justification of the induction principle is as follows. Statement (1) above says that $\mathcal{S}_0$ is true. By (2), if $\mathcal{S}_0$ is true, then $\mathcal{S}_1$ is true as well, so $\mathcal{S}_1$ is true. By (2) again, if $\mathcal{S}_1$ is true, then $\mathcal{S}_2$ is true as well, so $\mathcal{S}_2$ is true. One can continue applying (2) in this way to see that $\mathcal{S}_3, \mathcal{S}_4, ...$ are all true as well, so $\mathcal{S}_n$ is true for all natural numbers n.

For a formal proof of the induction principle, one uses the fact that the natural numbers are well-ordered, which means that any non-empty subset of the natural numbers has a least element. The proof is by contradiction. Suppose that $\mathcal{S}_n$ is not true for all $n \in \mathbb{N}$. Then the set $S := \{n | \mathcal{S}_n \text{ is false}\}$ is a non-empty subset of the $\mathbb{N}$ and thus has a least element, say k. Since k is minimal, $\mathcal{S}_n$ must be true for all $n < k$. In particular, $\mathcal{S}_{k-1}$ is true. But then statement (2) implies that $\mathcal{S}_k$ must be true as well. This contradicts $k \in S$, so $\mathcal{S}_n$ is true for all $n \in \mathbb{N}$.

Big O Notation

Constant Time Operations

An algorithm is said to be *constant time* or $O(1)$ time if its running time is bound by a value that does not depend on the size of the input.

Examples. Accessing an element of an array, swapping the values of two variables, and determining whether or not a number is even or odd are all $O(1)$ operations.