

Test II will be Monday March 28 from 1:00PM-2:00PM. It is closed book and closed note. It covers Chapter 3 Sections 1-5, Chapter 4 Section 3, Chapter 5 Section 1-3.

You may bring a SCIENTIFIC calculator. You may NOT use your cell phone or a laptop. You will not need a calculator as all numbers will be do-able by hand. You will be required to show your work.

Chapter 3

Section 1: Functions

You must be able to state and use the formal definitions of: function, one-to-one function, onto function, bijection, domain, codomain, and range.

You should be able to use the definitions above. For example, determine the domain and range of a function, determine if a rule is a function and if so whether it is one-to-one or onto.

You should know how to use and/or graph the following specific functions or ideas: the floor function, the ceiling function, the modulus operator, arrow diagrams, and the composition of functions.

You should be able to produce lots of examples satisfying (or not satisfying) the main definitions, such as give a rule that is not a function, give a function that is one-to-one function but not onto, or give a function that is onto but not one-to-one, and so forth.

Section 2: Sequences and Strings

You should know what a sequence (or subsequence) is and how to work with them (i.e. manipulate indices in convenient ways). You should know what it means for a sequence to be increasing, decreasing, nonincreasing, or nondecreasing and how to recognize these properties.

You should know what a string is and be familiar with the notation and language of strings such as length, substring, and concatenation.

Section 3: Relations

You need to know how to write, draw, and think about relations. You must know the definitions of the following terms: reflexive, symmetric, antisymmetric, transitive, inverse, and partial order. You should know how to show that a given relation does or does not have one of these properties. You should have lots of examples of relations that do or do not have these properties such as an example of a relation that is symmetric but not transitive.

Section 4: Equivalence Relations

You must know the definition of an equivalence relation and know how to use it. You should have examples of relations that are and are not equivalence relations. You should know what an equivalence class is, how to find the equivalence class of a particular element, how to find representatives of all equivalence classes for a particular equivalence relation, and that equivalence relations partition their underlying set. You should know that every partition also induces an equivalence relation.

Section 5: Matrices of Relations

You should know how to construct and use the matrix of a relation.

Chapter 4:

Section 3: Analysis of Algorithms

You must be able to state and use the formal definition of $f(n) = O(g(n))$, $f(n) = \Omega(g(n))$, and $f(n) = \theta(g(n))$. Given an $f(n)$ you must be able to determine its O , Ω or θ notation from the standard list of functions and to justify your answer. You should be able to find an $f(n)$ for the number of times a particular line in a simple program is executed and then identify (and justify) its O , Ω or θ notation. You should be able to prove simple statements concerning O , Ω or θ notation.

Chapter 5:

Section 1: Divisors

You need to be able to state and use the formal mathematical definition of what it means for d to divide n . You should know how to trace (or walk through) the standard algorithm for determining if an integer is prime. You should know how to use the previous algorithm to find the prime factorization of an integer. You should know how to determine the greatest common divisor or the least common multiple of two integers given their prime factorization.

Section 2:

You should know how to convert between numbers in different bases and you should be familiar with base 2 and base 16 in particular. You should know how to perform basic addition in different bases.

Section 3: The Euclidean Algorithm

You should know how to perform the Euclidean Algorithm on a pair of integers, tracing each iteration. You should know that the output is the greatest common divisor of the input. You should know how to use this information to write the greatest common divisor of two integers as a linear combination of the inputs.