



# BLM2502

# Theory of

# Computation

Spring 2017

# BLM2502 Theory of Computation

- » Course Outline
- » Week Content
- » 1 Introduction to Course
- » 2 Computability Theory, Complexity Theory, Automata Theory, Set Theory, Relations, Proofs, Pigeonhole Principle
- » 3 Regular Expressions
- » 4 Finite Automata
- » 5 Deterministic and Nondeterministic Finite Automata
- » 6 Epsilon Transition, Equivalence of Automata
- » 7 Pumping Theorem
- » 8 April 10 - 14 week is the first midterm week
- » 9 Context Free Grammars
- » 10 Parse Tree, Ambiguity,
- » 11 Pumping Theorem
- » 12 May 08 – 12 week is the second midterm week
- » 13 Turing Machines, Recognition and Computation, Church-Turing Hypothesis
- » 14 Turing Machines, Recognition and Computation, Church-Turing Hypothesis
- » 15 Review
- » 16 Final Exam date will be announced

# BLM2502 Theory of Computation

## » Every Thursday

> D111 13.00 – 15.50

## » Grading

> Final Exam 40%

> Midterm 20% each

> Homeworks 20 % (4 - 6)

## » Book

> Michael Sipser, Introduction to the Theory of Computation (3E), Thomson

> No handouts, its your responsibility to take notes.

# BLM2502 Theory of Computation

- » In this Theory of Computation course we will try to answer the following questions:
- » What are the mathematical properties of computer hardware and software?
- » What is a computation and what is an algorithm? Can we give rigorous mathematical definitions of these notions?
- » What are the limitations of computers? Can “everything” be computed? (As we will see, the answer to this question is “no”).
- » Purpose of the Theory of Computation:  
**Develop formal mathematical models of computation that reflect real-world computers.**

# BLM2502 Theory of Computation

- » Complexity Theory
- » The main question asked in this area is “What makes some problems computationally hard and other problems easy?”
- » Informally, a problem is called “easy”, if it is efficiently solvable. Examples of “easy” problems are
  - > Sorting a sequence of, e.g, 1,000,000 numbers,
  - > Searching for a name in a telephone directory
  - > Computing the fastest way to drive from Esenler to Beşiktaş.

# BLM2502 Theory of Computation

- » Complexity Theory
- » On the other hand, a problem is called “hard”, if it cannot be solved efficiently, or if we don’t know whether it can be solved efficiently. Examples of “hard” problems are
  - > Time table scheduling for all courses
  - > Factoring a 300-digit integer into its prime factors
  - > Computing a layout for chips in VLSI.
- » Central Question in Complexity Theory:  
**Classify problems according to their degree of “difficulty”. Give a rigorous proof that problems that seem to be “hard” are really “hard”.**

# BLM2502 Theory of Computation

- » Computability Theory
- » In the 1930's, scientists discovered that some of the fundamental mathematical problems cannot be solved by a "computer". (computers were invented in 1940s). For example: "Is an arbitrary mathematical statement true or false?"
- » To attack such a problem, we need formal definitions of the notions of
  - > computer
  - > algorithm
  - > computation

# BLM2502 Theory of Computation

- » Computability Theory
- » The theoretical models that were proposed in order to understand solvable and unsolvable problems led to the development of real computers.
- » Central Question in Computability Theory:

**Classify problems as being solvable or unsolvable.**

# BLM2502 Theory of Computation

- » Automata Theory
- » Automata Theory deals with definitions and properties of different types of “computation models”. Examples :
  - > Finite Automata. These are used in text processing, compilers, and hardware design.
  - > Context-Free Grammars. These are used to define programming languages and in Artificial Intelligence.
  - > Turing Machines. These form a simple abstract model of a “real” computer, such as your PC at home.
- » Central Question in Automata Theory:

**Do these models have the same power, or can one model solve more problems than the other?**

# BLM2502 Theory of Computation

## » Set Theory

» A set is a collection of well-defined objects.

- > The set of natural numbers is  $\mathbf{N} = \{1, 2, 3, \dots\}$ .
- > The set of integers is  $\mathbf{Z} = \{\dots, -3, -2, -1, 0, 1, 2, 3, \dots\}$ .
- > The set of rational numbers is  $\mathbf{Q} = \{m/n : m \in \mathbf{Z}, n \in \mathbf{Z}, n \neq 0\}$ .
- > The set of real numbers is denoted by  $\mathbf{R}$ .
- > If  $A$  and  $B$  are sets, then  $A$  is a subset of  $B$ , written as  $A \subseteq B$ , if every element of  $A$  is also an element of  $B$ .
- > If  $B$  is a set, then the power set  $P(B)$  of  $B$  is defined to be the set of all subsets of  $B$ :
  - ❖  $P(B) = \{A : A \subseteq B\}$ .
  - ❖ Observe  $\emptyset \in P(B)$  and  $B \in P(B)$ .

# BLM2502 Theory of Computation

## » Set Theory

- > If A and B are two sets, then
  - + their union is defined as
    - $A \cup B = \{x : x \in A \text{ or } x \in B\}$ ,
  - + their intersection is defined as
    - $A \cap B = \{x : x \in A \text{ and } x \in B\}$ ,
  - + their difference is defined as
    - $A \setminus B = \{x : x \in A \text{ and } x \notin B\}$ ,
  - + the Cartesian product of A and B is defined as
    - $A \times B = \{(x, y) : x \in A \text{ and } y \in B\}$ ,
  - + the complement of A is defined as
    - $\bar{A} = \{x : x \notin A\}$ .

# BLM2502 Theory of Computation

## » Set Theory

- > A binary relation on two sets  $A$  and  $B$  is a subset of  $A \times B$ .
- > A function  $f$  from  $A$  to  $B$ , denoted by  $f : A \rightarrow B$ , is a binary relation  $R$ , having the property that for each element  $a \in A$ , there is exactly one ordered pair in  $R$ , whose first component is  $a$ . We will also say that  $f(a) = b$ , or  $f$  maps  $a$  to  $b$ , or the image of  $a$  under  $f$  is  $b$ . The set  $A$  is called the domain of  $f$ , and the set  $\{b \in B : \text{there is an } a \in A \text{ with } f(a) = b\}$  is called the range of  $f$ .
- > A binary relation  $R \subseteq A \times A$  is an equivalence relation, if it satisfies the following three conditions:
  - +  $R$  is reflexive: For every element in  $a \in A$ , we have  $(a, a) \in R$ .
  - +  $R$  is symmetric: For all  $a$  and  $b$  in  $A$ , if  $(a, b) \in R$ , then  $(b, a) \in R$ .
  - +  $R$  is transitive: For all  $a$ ,  $b$ , and  $c$  in  $A$ , if  $(a, b) \in R$  and  $(b, c) \in R$ , then also  $(a, c) \in R$ .

# BLM2502 Theory of Computation

- » Boolean Logic
- » The Boolean values are 1 and 0, that represent true and false, respectively. The basic Boolean operations:
- » negation (or NOT), represented by  $\neg$ ,
- » conjunction (or AND), represented by  $\wedge$ ,
- » disjunction (or OR), represented by  $\vee$ ,
- » exclusive-or (or XOR), represented by  $\oplus$ ,
- » equivalence, represented by  $\leftrightarrow$  or  $\Leftrightarrow$ ,
- » implication, represented by  $\rightarrow$  or  $\Rightarrow$ .

# BLM2502 Theory of Computation

## » Truth Table

NOT	AND	OR	XOR	equivalence	implication
$\neg 0 = 1$	$0 \wedge 0 = 0$	$0 \vee 0 = 0$	$0 \oplus 0 = 0$	$0 \Leftrightarrow 0 = 1$	$0 \Rightarrow 0 = 1$
$\neg 1 = 0$	$0 \wedge 1 = 0$	$0 \vee 1 = 1$	$0 \oplus 1 = 1$	$0 \Leftrightarrow 1 = 0$	$0 \Rightarrow 1 = 1$
	$1 \wedge 0 = 0$	$1 \vee 0 = 1$	$1 \oplus 0 = 1$	$1 \Leftrightarrow 0 = 0$	$1 \Rightarrow 0 = 0$
	$1 \wedge 1 = 1$	$1 \vee 1 = 1$	$1 \oplus 1 = 0$	$1 \Leftrightarrow 1 = 1$	$1 \Rightarrow 1 = 1$

# BLM2502 Theory of Computation

- » Proof Techniques
- » In mathematics, a theorem is a statement that is true. A proof is a sequence of mathematical statements that form an argument to show that a theorem is true.
  - > axioms (assumptions about the underlying mathematical structures)
  - > hypotheses of the theorem
  - > previously proved theorems
  - > Lemmas
  - > Corollaries
- » There is no specified way of coming up with a proof, but there are some generic strategies that could be of help.

# BLM2502 Theory of Computation

## » Proof Techniques

## » Tips:

- > Read and completely understand the statement of the theorem to be proved. Most often this is the hardest part. Rewrite the statement in your own words.
- > Sometimes, theorems contain theorems inside them. For example, “Property A if and only if property B”, requires showing two statements:
  - > (a) If property A is true, then property B is true ( $A \rightarrow B$ ).
  - > (b) If property B is true, then property A is true ( $B \rightarrow A$ ).
- > Another example is the theorem “Set A equals set B.” To prove this, we need to prove that  $A \subseteq B$  and  $B \subseteq A$ . That is, we need to show that each element of set A is in set B, and that each element of set B is in set A.

# BLM2502 Theory of Computation

## » Proof Techniques

## » Tips:

- > Try to work out a few simple cases of the theorem just to get a grip on it (i.e., crack a few simple cases first).
- > Try to write down the proof once you have it. This is to ensure the correctness of your proof. Often, mistakes are found at the time of writing.
- > Finding proofs takes time, we do not come prewired to produce proofs. Be patient, think, express and write clearly and try to be precise as much as possible.

# BLM2502 Theory of Computation

## » Proof Techniques

- > Direct Proofs or Constructive Proofs or Proof by Construction
- > Nonconstructive Proofs
- > Proofs by Contradiction
- > Proofs by Induction
- > Pigeon Principle

# BLM2502 Theory of Computation

- » Proof Techniques - Direct Proofs
- » As the name suggests, in a direct proof of a theorem, we just approach the theorem directly.
- » **Theorem:** If  $n$  is an odd positive integer, then  $n^2$  is odd as well.
- » **Proof:** An odd positive integer  $n$  can be written as:  
 $n = 2k + 1$ , for some integer  $k \geq 0$ . Then:
  - ❖  $n^2 = (2k + 1)^2 = 4k^2 + 4k + 1 = 2(2k^2 + 2k) + 1$ .
  - ❖ Since  $2(2k^2 + 2k)$  is even, and “even plus one is odd”, we can conclude that
  - ❖  $n^2$  is odd.

# BLM2502 Theory of Computation

## » Proof Techniques - Constructive Proofs (Proof By Construction)

- » Many theorem states that a particular type of object exists
- » One way to prove is to find a way to construct one such object
- » This technique is called proof by construction
- » **Theorem:** There exists a rational number  $p$  which can be expressed as  $q^r$ , with  $q$  and  $r$  both irrational.
- » Find  $p, q, r$  satisfying the above condition
- » **Proof:**  $\sqrt{2}^{\sqrt{2}^{\sqrt{2}}} = ?$
- »  $p, r = \sqrt{2}$  satisfies or not?
- »  $P = \sqrt{2}^{\sqrt{2}}, r = \sqrt{2}$  satisfies or not?

# BLM2502 Theory of Computation

- » Proof Techniques - Nonconstructive Proofs
- » In a nonconstructive proof, we show that a certain object exists, without actually creating it. Here is an example of such a proof:
- » **Theorem:** There exist irrational numbers  $x$  and  $y$  such that  $xy$  is rational.
- » **Proof:** There are two possible cases.
- » Case 1:  $\sqrt{2} \in \mathbb{Q}$ .
- » In this case, we take  $x = y = \sqrt{2}$ . In Theorem below, we will prove that  $\sqrt{2}$  is irrational.

# BLM2502 Theory of Computation

- » Proof Techniques - Nonconstructive Proofs
- » Case 2:  $\sqrt{2}^{\sqrt{2}} \notin \mathbb{Q}$ .
- » In this case, we take  $x = \sqrt{2}$  and  $y = \sqrt{2}$ . Since
$$x^y = \sqrt{2}^{\sqrt{2}^{\sqrt{2}}} = \sqrt{2}^2 = 2,$$
- » the claim in the theorem follows.
- » This proof indeed proves the theorem, but it does not give an example of a pair of irrational numbers  $x$  and  $y$  such that  $x^y$  is rational.

# BLM2502 Theory of Computation

- » Proof Techniques - Proofs by Contradiction
- » The proof by contradiction is grounded in the fact that any proposition must be either true or false, but not both true and false at the same time.
- » One common way to prove a theorem is to assume that the theorem is false, and then show that this assumption leads to an obviously false consequence (also called a contradiction)
- » This type of reasoning is used frequently in everyday life, as shown in the following example

# BLM2502 Theory of Computation

- » Proof Techniques - Proofs by Contradiction
- » Let us define a number is rational if it can be expressed as  $p/q$  where  $p$  and  $q$  are integers; if it cannot, then the number is called irrational.
- » **Theorem:**  $\sqrt{2}$  (the square root of 2) is irrational.
- » **Proof:** Assume that  $\sqrt{2}$  is rational. Then, it can be written as  $p/q$  for some positive integers  $p$  and  $q$ .
- » In fact, we can further restrict that  $p$  and  $q$  does not have common factor.
- » If  $D$  is a common factor of  $p$  and  $q$ , we use  $p' = p/D$  and  $q' = q/D$  so that  $p'/q' = p/q = \sqrt{2}$  and there is no common factor between  $p'$  and  $q'$
- » Then, we have  $p^2/q^2 = 2$ , or  $2q^2 = p^2$

# BLM2502 Theory of Computation

## » Proof Techniques - Proofs by Contradiction

- » Since  $2q^2$  is an even number,  $p^2$  is also an even number
- » This implies that  $p$  is an even number (why?)
- » So,  $p = 2r$  for some integer  $r$   $2q^2 = p^2 = (2r)^2 = 4r^2$
- » This implies  $2r^2 = q^2$
- » So,  $q$  is an even number
- » Something wrong happens!.. We now have:
  - » “ $p$  and  $q$  does not have common factor”
  - » AND
  - » “ $p$  and  $q$  have common factor”
- » **This is a contradiction**
- » Thus, the assumption is wrong, so that  $\sqrt{2}$  is irrational

# BLM2502 Theory of Computation

- » Proof Techniques - Proofs by Induction
- » For each positive integer  $n$ , let  $P(n)$  be a mathematical statement that depends on  $n$ . Assume we wish to prove that  $P(n)$  is true for all positive integers  $n$ . A proof by induction of such a statement is carried out as follows:
- » Basis: Prove that  $P(1)$  is true.
- » Induction step: Prove that for all  $n \geq 1$ , the following holds: If  $P(n)$  is true, then  $P(n + 1)$  is also true.
- » In the induction step, we choose an arbitrary integer  $n \geq 1$  and assume that  $P(n)$  is true; this is called the induction hypothesis. Then we prove that  $P(n + 1)$  is also true.

# BLM2502 Theory of Computation

- » Proof Techniques - Proofs by Induction

- » **Theorem:** For all positive integers  $n$ , we have

$$1 + 2 + 3 + \dots + n = n(n + 1)/2$$

- » **Proof:** Start with the basis of the induction. If  $n = 1$ , then the left-hand side is equal to 1, and so is the right-hand side. So the theorem is true for  $n = 1$ .

- » For the induction step, let  $n \geq 1$  and assume that the theorem is true for  $n$ , i.e., assume that

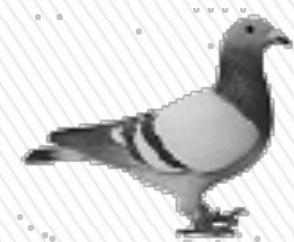
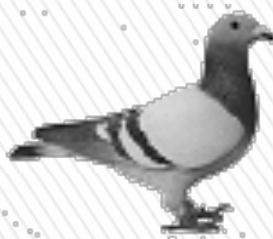
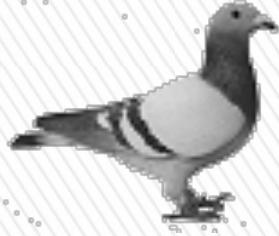
$$1 + 2 + 3 + \dots + n = n(n + 1)/2$$

- » We have to prove that the theorem is true for  $n + 1$

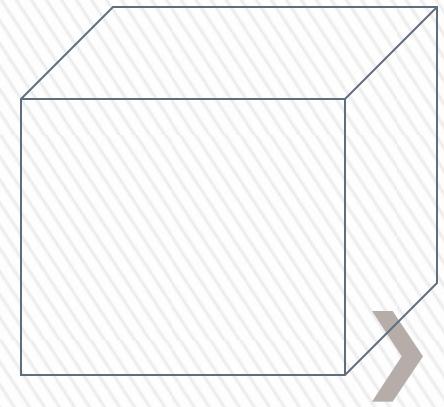
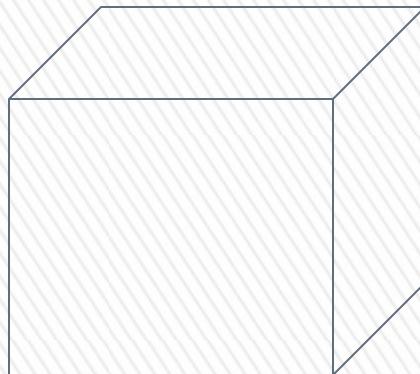
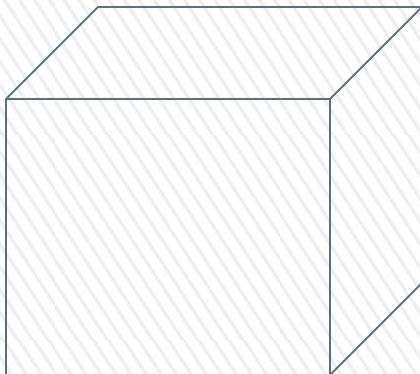
# BLM2502 Theory of Computation

- » Proof Techniques – Pigeon Principle
- » If  $n + 1$  or more objects are placed into  $n$  boxes, then there is at least one box containing two or more objects.
- » In other words, if  $A$  and  $B$  are two sets such that  $|A| > |B|$ , then there is no one-to-one function from  $A$  to  $B$ .

4 pigeons

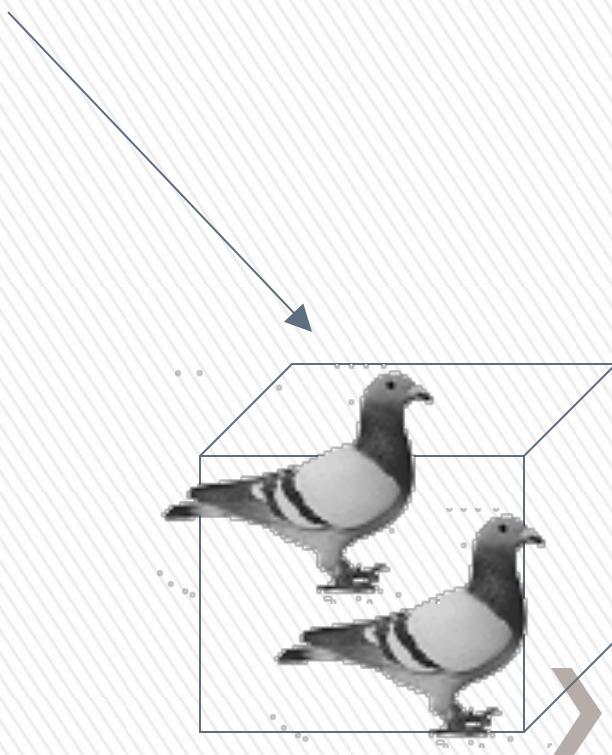
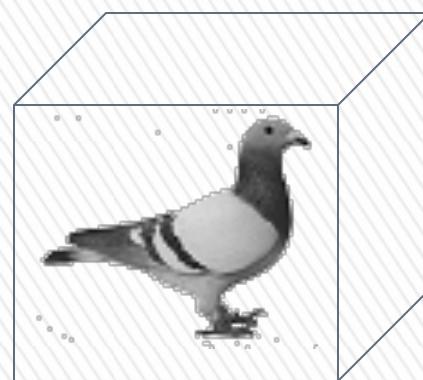
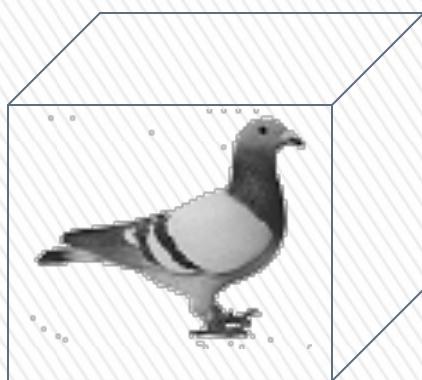


3 pigeonholes





A pigeonhole must  
contain at least two pigeons



# BLM2502 Theory of Computation

- » Proof Techniques – Pigeon Principle
- » **Theorem:** Let  $n$  be a positive integer. Every sequence of  $n^2 + 1$  distinct real numbers contains a subsequence of length  $n + 1$  that is either increasing or decreasing.
- » **Proof:** For example consider the sequence (20, 10, 9, 7, 11, 2, 21, 1, 20, 31) of  $10 = 3^2 + 1$  numbers. This sequence contains an increasing subsequence of length  $4 = 3 + 1$ , namely (10, 11, 21, 31).

# BLM2502 Theory of Computation



# BLM2502 Theory of Computation

Let  $(a_1, a_2, \dots, a_{n^2+1})$  be an arbitrary sequence of  $n^2 + 1$  distinct real numbers. For each  $i$  with  $1 \leq i \leq n^2 + 1$ , let  $inc_i$  denote the length of the longest increasing subsequence that starts at  $a_i$ , and let  $dec_i$  denote the length of the longest decreasing subsequence that starts at  $a_i$ . Using this notation, the claim in the theorem can be formulated as follows:

There is an index  $i$  such that  $inc_i \geq n + 1$  or  $dec_i \geq n + 1$ .

Claim can be proved by using contradiction and pigeon hole principle :

Assume that  $inc_i \leq n$  and  $dec_i \leq n$  for all  $i$  with  $1 \leq i \leq n^2 + 1$ . Consider the set

$$B = \{(b, c) : 1 \leq b \leq n, 1 \leq c \leq n\},$$

and think of the elements of  $B$  as being boxes. For each  $i$  with  $1 \leq i \leq n^2 + 1$ , the pair  $(inc_i, dec_i)$  is an element of  $B$ . So we have  $n^2+1$  elements  $(inc_i, dec_i)$ , which are placed in the  $n^2$  boxes of  $B$ .

# BLM2502 Theory of Computation

By the pigeon hole principle, there must be a box that contains two (or more) elements. In other words, there exist two integers  $i$  and  $j$  such that  $i < j$  and

$$(inc_i, dec_i) = (inc_j, dec_j).$$

Recall that the elements in the sequence are distinct. Hence,  $a_i \neq a_j$ . We consider two cases:

First assume that  $a_i < a_j$ . Then the length of the longest increasing subsequence starting at  $a_i$  must be at least  $1 + inc_j$ , because we can append  $a_i$  to the longest increasing subsequence starting at  $a_j$ . Therefore,  $inc_i \neq inc_j$  which is a contradiction.

The second case is when  $a_i > a_j$ . Then the length of the longest decreasing subsequence starting at  $a_i$  must be at least  $1 + dec_j$ , because we can append  $a_i$  to the longest increasing subsequence starting at  $a_j$ . Therefore,  $dec_i \neq dec_j$  which is a contradiction.