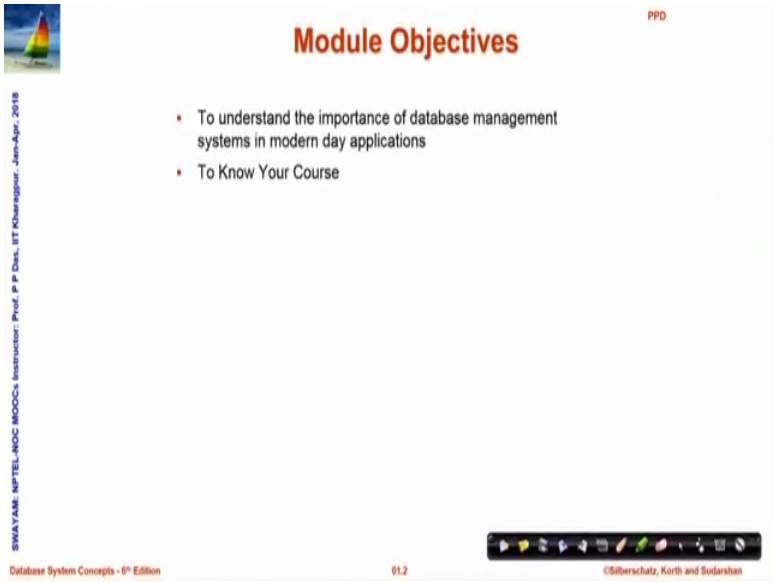
Database Management System
Prof. Partha Pratim Das
Department of Computer Science & Engineering
Indian Institute of Technology, Kharagpur

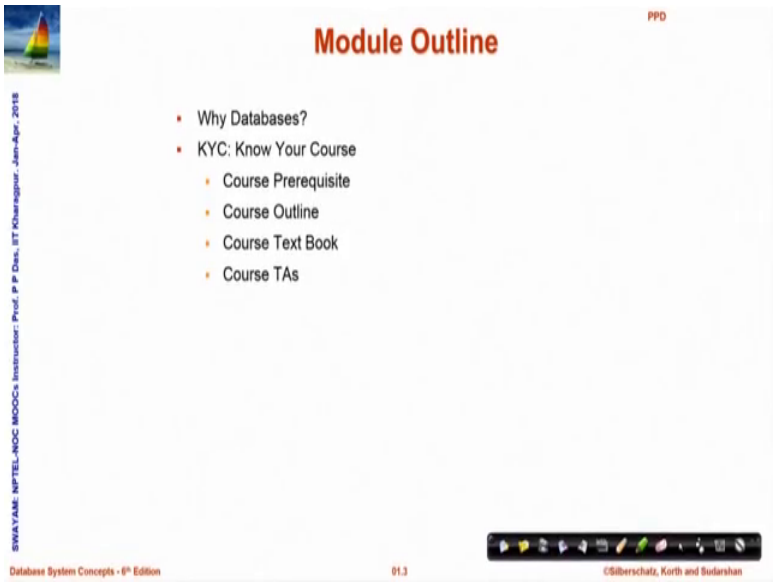
Lecture - 01
Course Overview

(Refer Slide Time: 00:45)



Welcome to database management systems. In this course, we will have 40 modules; each module would be of about half an hour. So, this is the first module, where we would talk about the overview of the course. So, we will discuss the importance of database management systems in modern day applications, and we will familiarize you with different aspects of the course.

(Refer Slide Time: 01:06)

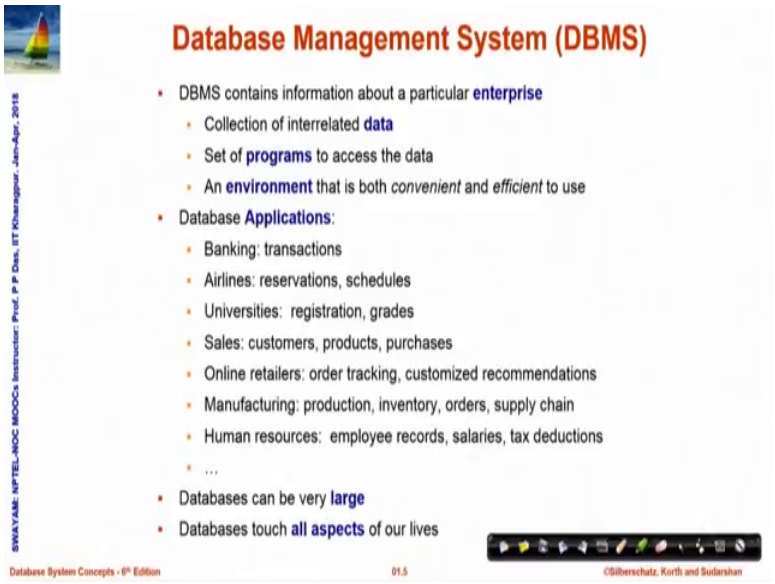


So, this will be the outline. First, we will try to explain why we need databases, and then we will run through a KYC on the course prerequisites, course outline, the textbook and the TAs who will help us in this course.

(Refer Slide Time: 01:28)

So, first why do we need databases?

(Refer Slide Time: 01:32)



A database management system contains information about a particular enterprise. So, it deals with collection of interrelated data. We have a set of programs which access and manipulate that data; and together a DBMS presents an environment for convenient as well as efficient use of data of the enterprise under consideration. So, there could be different database application.

Actually, if we look around in the world that we are living today, every aspect of our life has been touched by certain database application. Some of the very common and wide applications will include banking; we are doing net banking very regularly. Net banking is highly distributed database application that allow us to do different kinds of retail as well as corporate banking.

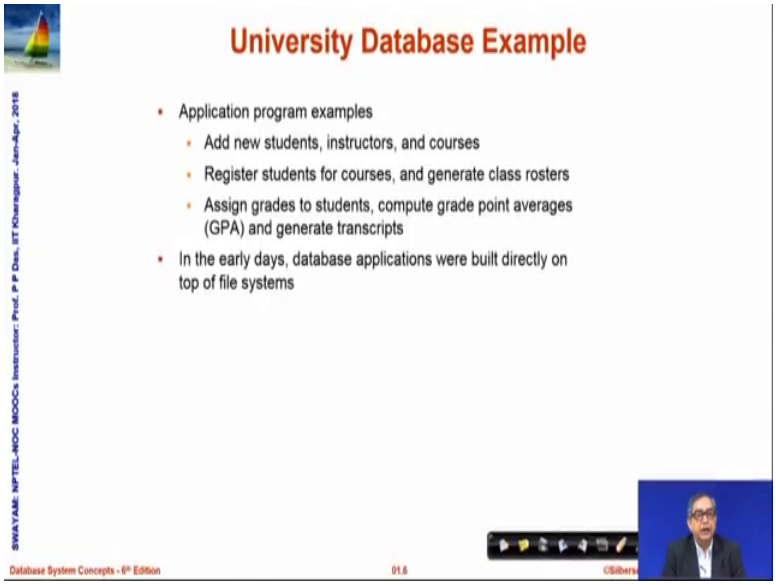
We perform different kind of booking reservation, railway reservation, airline reservation, hotel reservations all of these are now possible over internet as well these are all big database applications. When we are attending colleges, universities, the students, courses, teachers course enrolments, performance of the students, and different courses, examination all are parts of huge university database applications.

There are different sales applications, online retailing applications like I am sure all of you have been using some of the Amazon, Flipkart, E-bay, Snapdeal, all these are online retail database applications which allow us to order to select items to make payments and to track the deliveries of different items that we have ordered. There are database

applications in terms of manufacturing, production systems, inventory management, any big factory of manufacturing need to use a huge database application to manage the supply chain, the inventory, the orders everything. There are database applications in human resources applications like Linkedin are huge human resource application of database, the social media applications all involve different kinds of database applications.

So, in this database management system course, we are going to understand how such applications can be designed developed and managed over a period of time. Database applications are typically characterized by the fact that they are very large. If we have a relatively small set of data then possibly we will be able to manage it in terms of using an excel sheet or a couple of excel sheets, but soon when it goes beyond a certain size we need a database application. So, the fact of being large is a critical factor of any DBMS. So, with all this we as we observe the database says cover all aspects of our life and hence understanding DBMSs is a critical requirement for any computer science information technology student.

(Refer Slide Time: 05:54)

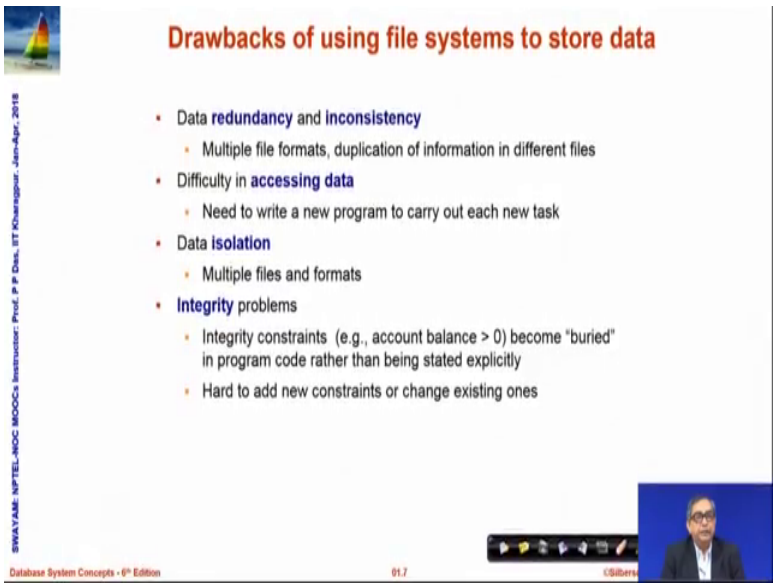


So, if we talk about a specific database example say university database, then we will have several application programs which will allow us to do several required applications like when new students join, we would need to add the students. We will need to add new courses when courses are floated; we will need to add new instructors

when faculty join; we will need to do allotments; we will need to do registration of students for courses. We will need to conduct examinations, we will need to assign grades to students when they are graded for different courses and compute their GPA and so on. So, all the activities that a university has to deal with are application programs of varied kinds that the university database need to work with.

In the earlier days, before the days of databases, typically such applications used to be managed through file systems. We all know that all our system has a file system, where different files text as well as data files can be stored, and information can be written in these files in a certain order. And just to quickly recall file systems typically have a large number of sequential files which can be written and read in a certain order and some random access files where you can reach a particular point in the file to do certain access operation certain manipulation operations. So, in the earlier days, it was a collection of file systems which managed large enterprise data as is required.

(Refer Slide Time: 07:53)



But over time it was observed that the file systems to store data to manage data has a lot of drawbacks. For example, if you look at in a file system, there is often a lot of data redundancy and inconsistency. These are terms which we will loosely define here and as we go across along the course, we will understand these terms better. But redundancy just to explain redundancy is a concept where the same data is written at multiple places in different forms and that may give rise to several forms of inconsistencies because if

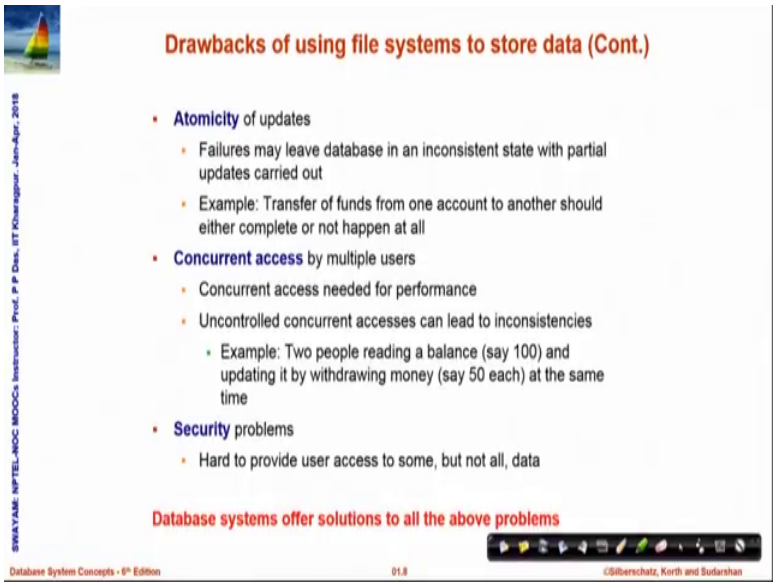
you write the same data in multiple files, because you need to deal with many of these aspects.

So, there is a file for students, there is a file for teachers, there is a file for particular courses, there is a file for enrolment and so on several data items may be redundantly copied in multiple files. And once that is done then we can it is very easy to get inconsistent in terms of the data because you may update the data in one file, and may forget to update the data in another file, so that is one of the first problems with using the file systems.

Then this difficulty in accessing the data because as I said the data is the files often are sequential in nature, even if they are random access, then every task might need to use data from multiple files. And opening those files and reaching to the appropriate point of access is a non-trivial task. Then there is issues of data isolation, because there are multiple types of files, there are multiple formats used therein. Very importantly there are a lot of integrity problems, the database any database application need to have a lot of integrity.

For example, if you want to withdraw money from an account - bank account, then certainly the balance needs to be positive, you can withdraw only that much amount up to that much amount, which exists in the account. So, any application will need to check for this. So, if you use a file system based application to store the data then at every point wherever you are updating the balance, you will need to make such checks which make the application quite complicated, and often creates the possibility that certain integrity checks may be missed out. So, it is hard to code these new constraints over a period of time.

(Refer Slide Time: 10:40)



Then there are issues of atomicity of update. What atomicity means is the ability to do certain operations in a as a single unit. So, what you want is either that operation happens or the and if it happens then it happens in full in totality, whereas otherwise it may not happen at all. For example, consider that there is a funds being transformed from one account to another, so this means the account from which the funds are being transferred needs to be debited certain amount, and that same amount has to get credited to the accounts to which it is being paid.

Now, if for some reason of failure or because of the fact that there was link issues or something, if you are not able to make this whole transfer, then it is possible that you have already debited the account, but you have not been able to create that account. Now, this will be a major cause of inconsistency in the database. So, what you want is if the transfer can happen then it must happen in totality that is what that debit and the credit must happen together or nothing should happen at all. So, there are several examples of requirement of automaticity for a update which is critical for maintaining consistency of the data.

The other aspect which has become very, very deeply required in every aspect of database is concurrency of access. If there is a database then certainly there is not a single user, there are multiple hundreds of users you think about net banking, you think about railway reservation, multiple users are trying to make bookings in multiple trains

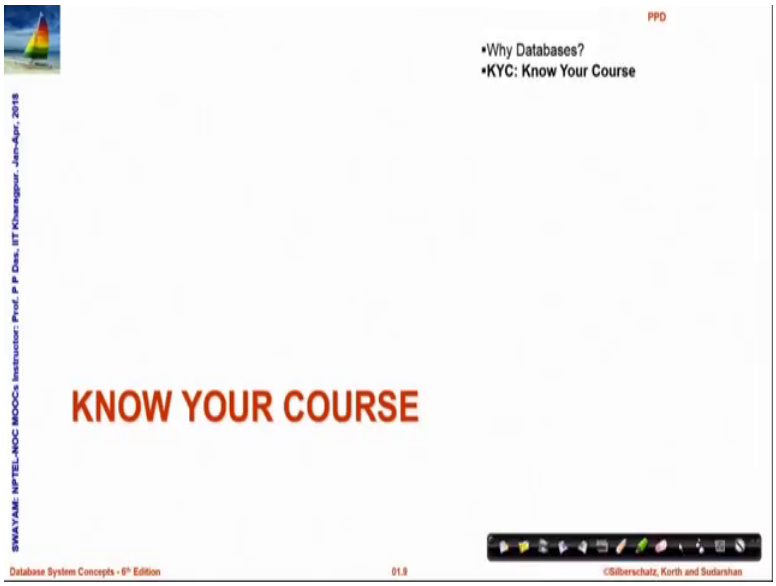
from varied stations to vary stations in different classes and so on. So, all of this must go on at the same time that is what is called the concurrency of update.

So, it is quite possible that while you are trying to update, you check berth availability on a certain train on a certain date that you intend to travel. At the same time, someone else may be checking for the berth availability on the same train on the same date, and there could be conflict of concurrency because there may be one berth available and you are trying to book that you have seen that one berth is available.

So, you go ahead and book it try to book it, and there is another user who also saw that one berth is available and that user makes payments and tries to book that. So, concurrency needs to make sure that both of these users should not be allowed to make the booking to the same berth because then that will be a disaster. So, uncontrolled concurrency can add to several inconsistencies in the application.

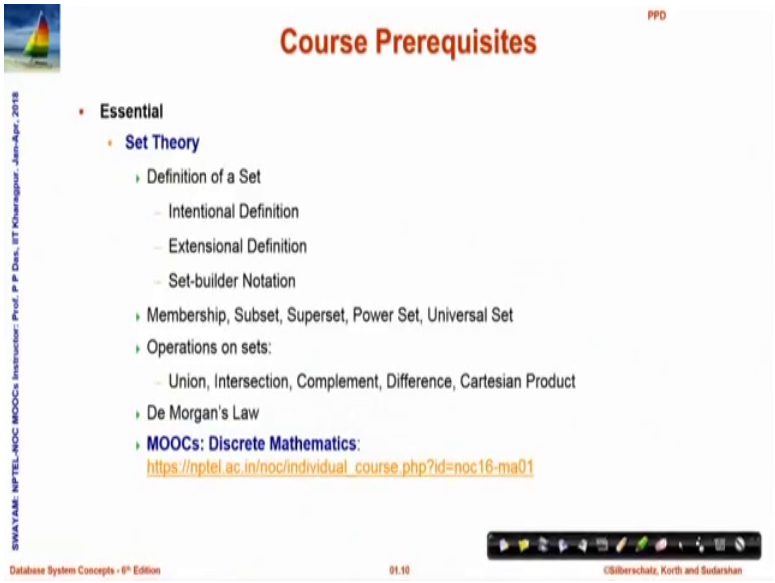
Then certainly there are you all would be very familiar that today we are living under a whole lot of security threats. So, there has to be proper security that it should be possible to access the data by a user to the extent the user is allowed to do that. So, as a user you should be able to access certain parts of the data, the manager of the system, the administrator should be able to access a bigger part of the data possibly. So, security is hard to provide in terms of a file system based applications to store data. So, all these with we conclude the data based systems had those which provide solution to take care against all those above problems, and we are trying to learn how to do such things.

(Refer Slide Time: 14:37)



So, moving on I would quickly take you through familiarizing with you with the overall plan of the course.

(Refer Slide Time: 14:47)

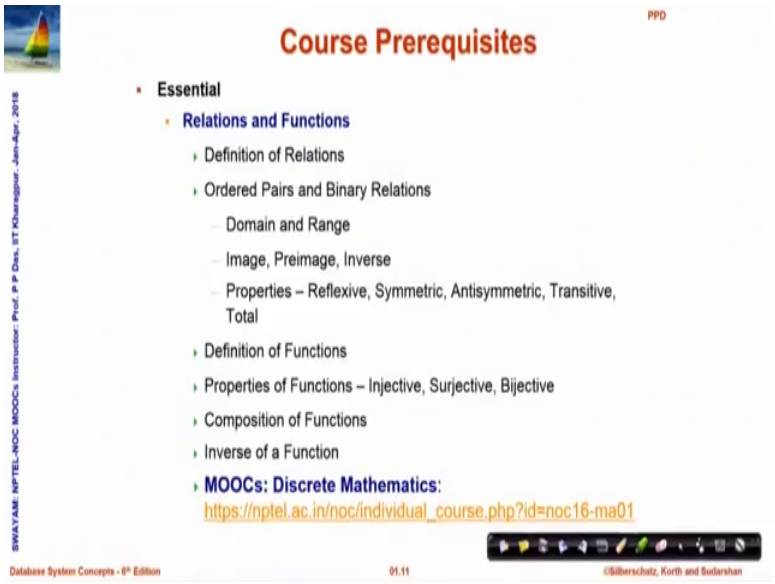


So, what we first I talk about are the course prerequisites. These prerequisites are kind of certain elementary level knowledge in computer science and related discrete mathematics that you should have. You should be that would make it easier for you to understand and follow the course; otherwise if you find at any stage that you are finding any of this aspect difficult to follow in the course then I would advise that you go back to

some of the background material and try to study them. So, I have tried to list down these prerequisite topics, one certainly is the set theory because that is the basic premise on which databases are designed on.

So, starting from definition of the set membership, concepts of subset, superset, power set, different operations of union, intersection, complementation, difference Cartesian product De Morgans law, all these basic set theory you should be very familiar and conversant with. If you are not I have mentioned one MOOC's course this is a past course, but you can access the videos and the contents which has a very nice discussion on these aspects of set theory which you may refer to.

(Refer Slide Time: 16:13)

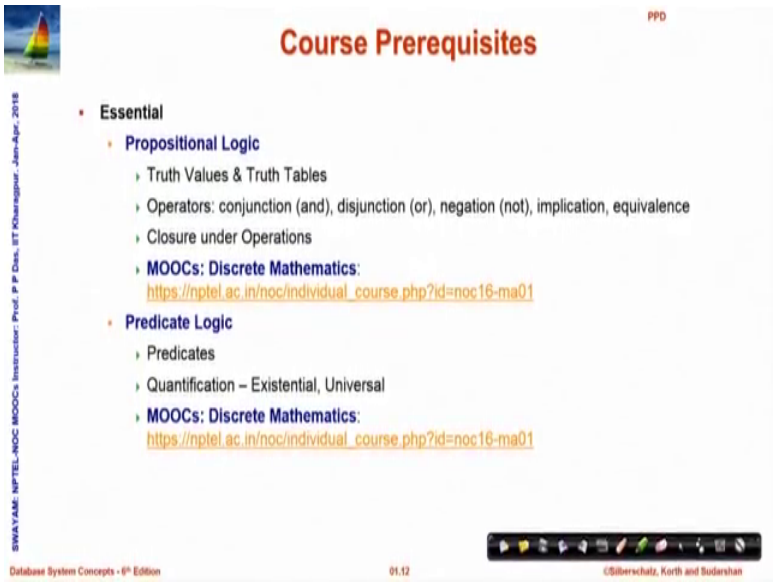


Moving on the next, which goes on top of the sets are the concept of relations and functions. We all know that a relation is a subset of a set. So, if I have a set A, then I can define a binary relation over that set A, which is basically the pair of elements from set A or in other words the relation binary relation is a subset of the cross product of A with itself where the domain and the range are related together. So, there are concepts of the image of a domain the pre-image of the range the inverse relation, and several basic properties of relations like the relation being reflexive, symmetric, anti-symmetric, transitive, total relations and so on.

You should be familiar with these; otherwise, you will find difficult to follow major concepts in the database systems because this tribal system primarily the one that we are

going to take you through in this course is relational in model. So, it is heavily based on relations and functions. And you should understand one specifically a relation becomes a function, and when what is meant by functions being injective, subjective, bijective, what is meant by composition, and inverse of functions and so on. Again if you need, you can refer to this MOOC's course on discrete mathematics to brush up your knowledge about relations and functions.

(Refer Slide Time: 17:52)

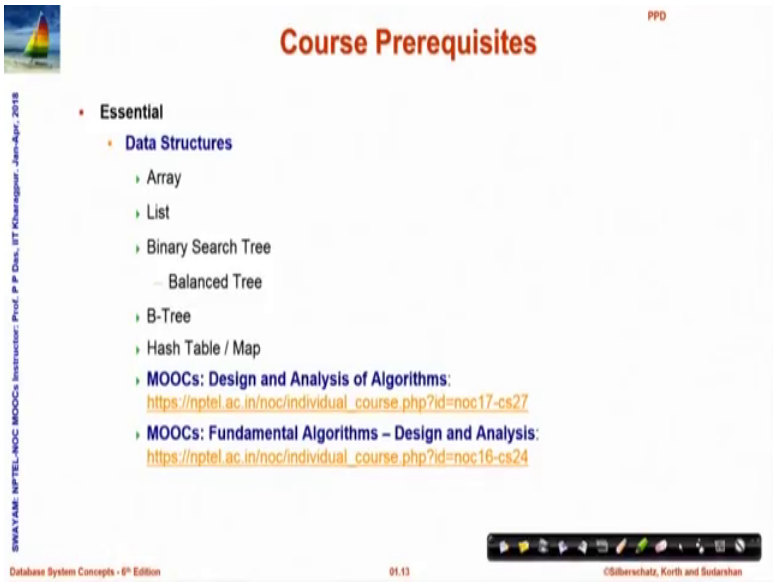


We also need you to have a basic understanding of the propositional logic which is truth values true and false; and the different operations of conjunction, disjunction, negation that is and or not, what is meant by implication, what is meant by equivalence. You know that given two variables, which have or two propositions which can take a value true or false the conjunction of them can be represented in terms of a truth table where we say that only when both these propositions are true, then the resultant conjunctive proposition becomes true; otherwise, a conjunctive proposition is false. So, you should be familiar with these concepts; if you are not, please brush up your ideas about propositional logic.

We need a little bit of predicate logic as well a predicate logic in contrast to propositional logic deals with quantification that the knowledge of existential and universal quantifier. When you say that whether certain proposition predicates hold for all values in the domain or for some value in the domain whether there exists some value for which it

holds or whether for all values it holds. And based on that predicate logic is build up we do not need very advanced concept here just basic level familiarization will help and the same MOOC's course on discrete mathematics would be of your help in case you need to brush it up further.

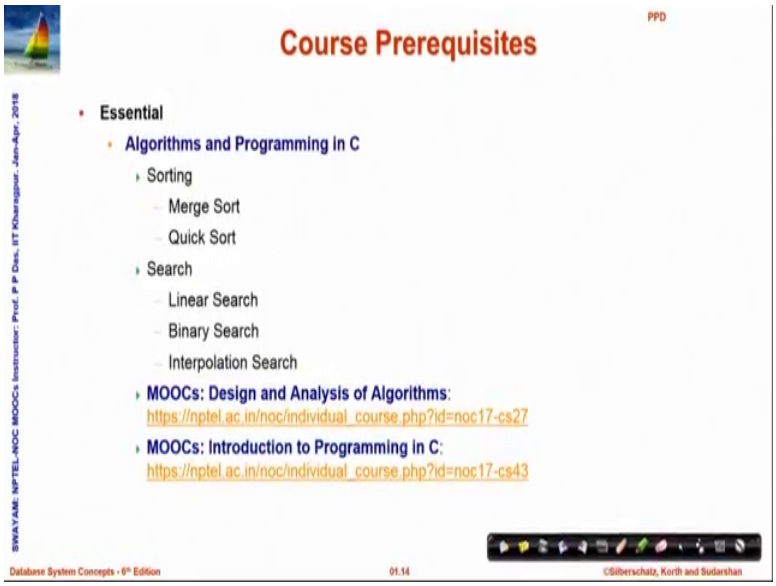
(Refer Slide Time: 19:33)



On aspects of computer science, certainly you need a good familiarity with the data structures array list particularly binary search tree, what is called a binary search tree, what is meant by a height of a binary search tree, when we say that a binary search tree is balanced. What are the ways and conditions of balancing is particularly the B-trees for organizing good search trees hash tables, what is hashing we need you to be familiar with this concept, because the databases will be heavily designed based on the concepts of b trees and hash tables and so on.

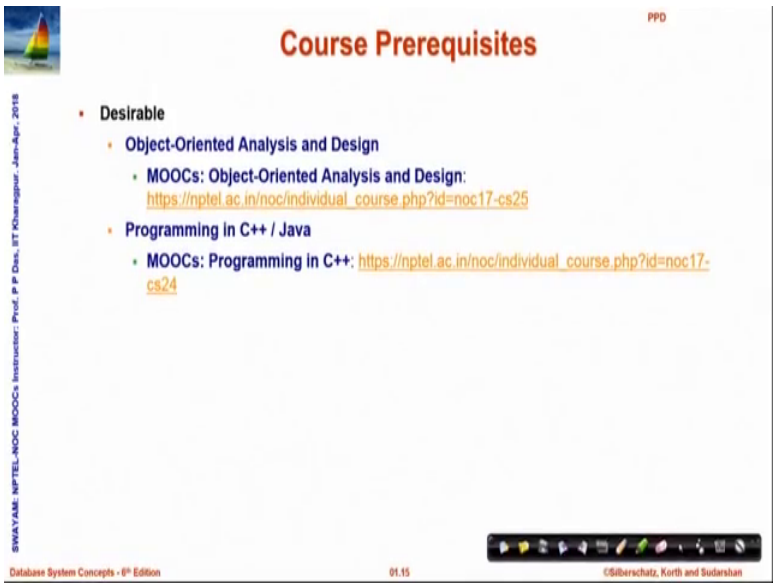
There are courses on design and analysis of algorithms, fundamental of algorithms have mentioned two excellent courses in MOOC's from which you can brush up if you need to.

(Refer Slide Time: 20:31)



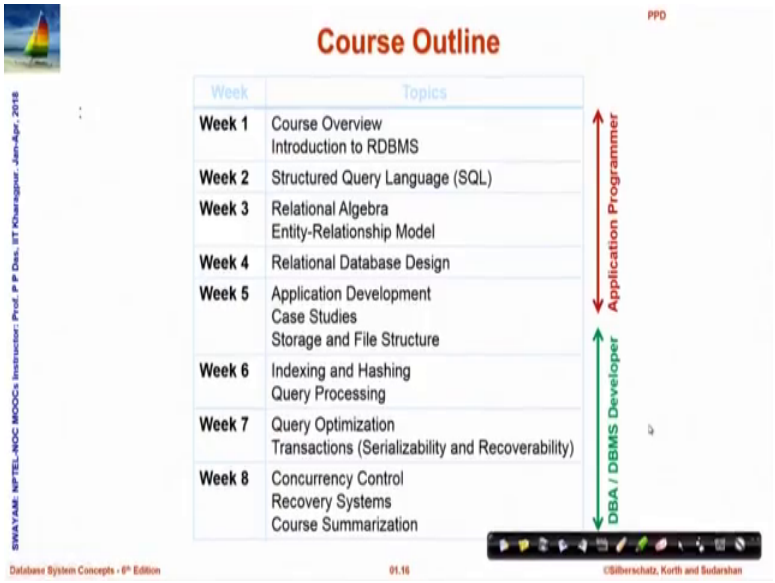
Certainly you need certain familiarity with common algorithms particularly I had mentioned sorting and searching algorithms, because these are critical for database applications. And again the same MOOC's courses would be of your help. And it will be good to have familiarity with programming in C, because several of the applications need some application high level application programming to be performed. And we would assume that those aspects we described in C because that is a fundamental and most commonly known language.

(Refer Slide Time: 21:11)



Besides these prerequisites which I have marked as essential, because they will certainly be required for the major parts of the course, it will be good if you have some familiarity with object oriented analysis; and design and some language which is more heavily object oriented object aligned in nature like C plus plus or java. So, again some MOOCs courses the related MOOCs courses are mentioned here in case you need to brush up.

(Refer Slide Time: 21:46)



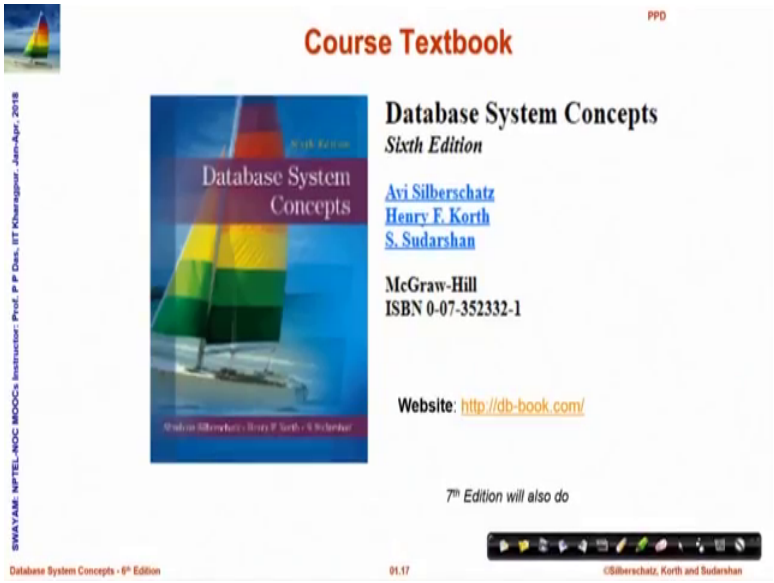
Moving on this is your course outline. So, as I said the course comprise 40 modules. So, it is divided into 8 weeks. So, this plan is given based on what we do in different weeks from week 1 to week 8. So, as the course unfolds, you will be able to we will take you through these modules on these topics. And on the right, you can find that I have marked that the initial part of the course which we cover from week one to first half of week 5 is primarily meant for application programming which it means that the database system has already been designed and basic premises the schemas and constraints have been set up.

But now you want to write different data query different data manipulation applications and that is where the large volume of database engineers work. So, they are called application programmers, so that is I should say the first level in terms of a database understanding. And you must target to become a master of application programming to get started with.

The other half of the course which start from the middle of week five with the storage and file structure and goes on till the next is meant for the analysts who are responsible either for designing a particular database which the application programmer can use tune that for performance index it properly design queries to be efficient. So, these kind of analysts will be involved the more with the understanding of the second part of the course.

And the second part of the course would also be useful for the database a DBMS designers. If you want to really become an advanced programmer we want to work on database engineering in terms of creating database management systems not merely creating databases or database applications, then you need to have a good initial command over the second half of the course. So, while you prepare for the course where you go through the modules please keep this in mind that your familiarity with the application programming must be at the highest level. And in the later parts will be relatively little advanced, but they are required for good design and good development of consistent efficient system in future.

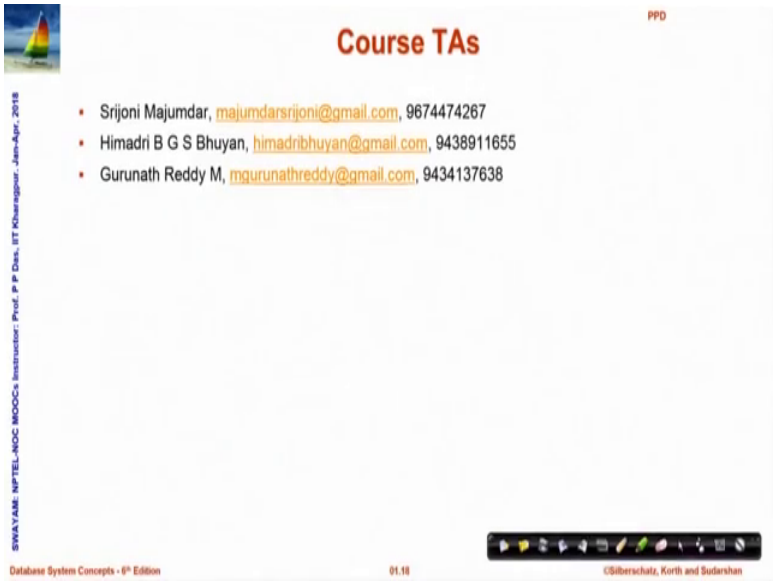
(Refer Slide Time: 24:39)



We will follow a textbook. This is as you can see this is a sixth edition that I am following in this course. It is called Database System Concept by Silberschatz, Korth and Sudarshan. This is kind of a classical book in database systems; current version actually is the seventh edition. So, if you get access to the seventh edition, you can use that as

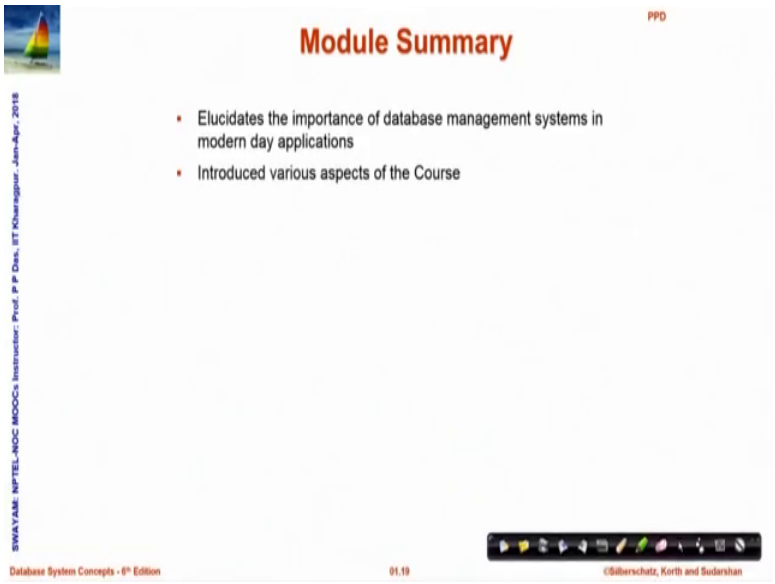
well, but whatever we are following in this course sixth edition is good enough. So, I advise the, that you try to get a copy of this book to yourself.

(Refer Slide Time: 25:24)



So, moving on we have different three TAs we will help you in this course Srijoni Majumdar, Himadri Bhushan Bhuyan and Gurunath Reddy. So, these are their emails; I have also put their mobile numbers. However, I would advise that unless you are really stuck avoid calling them on the mobile, because they are research students as well, busy with their own work as well, but they can certainly put all your questions on the forum, which will be promptly responded by some of these TAs or by myself. And in case you would have very specific follow ups to do, you can write email to one or all of these TAs. So, this is about the course overview.

(Refer Slide Time: 26:17)



So, in this module we have discussed about the importance about database management systems in the modern day applications. And we have tried to familiarize you with different aspects of the course. And I reiterate that please give due consideration to the prerequisites as mentioned, we have floated in an assignment called assignment zero where there are questions on different aspects of these prerequisites, please try to solve that assignment and see your performance. This assignment I would like to mention that this assignment will not go in the final evaluation of the performance of the course; this is just to give you an idea for self assessment of your preparedness for this course.

So, if you find that on questions on certain topics say on relation function or on data structure, if you have not been able to answer the questions well then it will be good to check back and go through the prerequisites once more. But please keep in mind that database management system is a course which depends on these background knowledge quite heavily. So, if you have gaps in understanding those topics, then all through the course you will get into several difficulties in understanding and problem solving. So, that is about our first module. So, from the next module, we will start introducing the database management system. Enjoy the course, and try to learn, try to become a good database engineer.