

Computational Commutative Algebra
Prof. Manoj Kummini
Department of Mathematics
Chennai Mathematical Institute

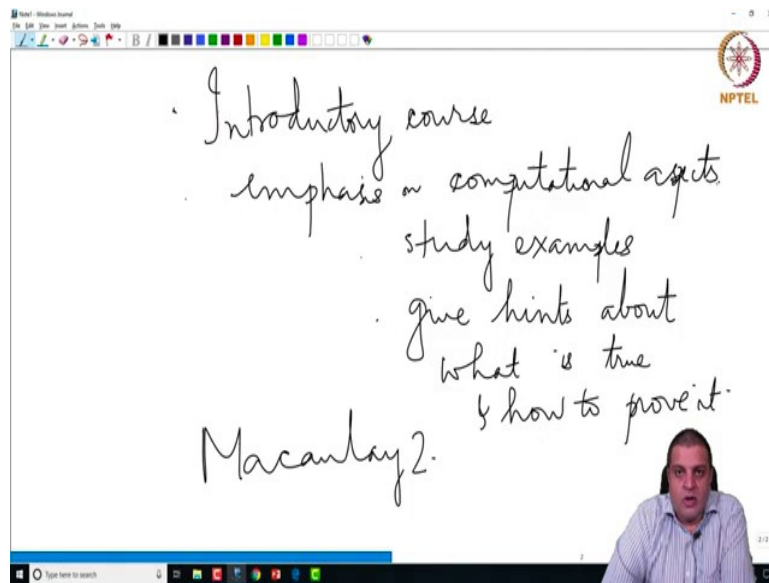
Lecture – 01
Definitions

(Refer Slide Time: 00:21)



Hello, welcome to this course on Computational Commutative Algebra. I am Manoj Kummini and I teach at Chennai Mathematical Institute. So, what is this course about? This is meant to be an introductory course in commutative algebra.

(Refer Slide Time: 00:59)



However, there will be an emphasis on computational aspects. Hence, in addition to doing the typical topics that are taught in a first course in commutative algebra, we will also try to understand the algorithms behind computing things with about doing some modules using a computer.

And therefore, some of the theorems that we prove in these lectures may not be as general as they are typically taught in a commutative algebra course. However, we will try to understand what is going on in a computer, when it tries to compute various things about ideals or modules.

So, why do we care about computing things? It is an easy way to study examples, to construct examples. And also sometimes give an idea on how to solve problems. This is the reason why there is an increasing interest in learning computational techniques in commutative algebra and similarly with an intersection with algebraic geometry.

So, all of these things we will learn using an computer algebra language called Macaulay2; which is freely downloadable from the web.



(Refer Slide Time: 03:07)

References:

- Cox, Little and O'Shea: *Ideals, Varieties and Algorithms*, Springer.
- Eisenbud: *Commutative Algebra with a View towards Algebraic Geometry*, Springer.
- Reid: *Undergraduate Commutative Algebra*, Cambridge University Press.

Macaulay2:

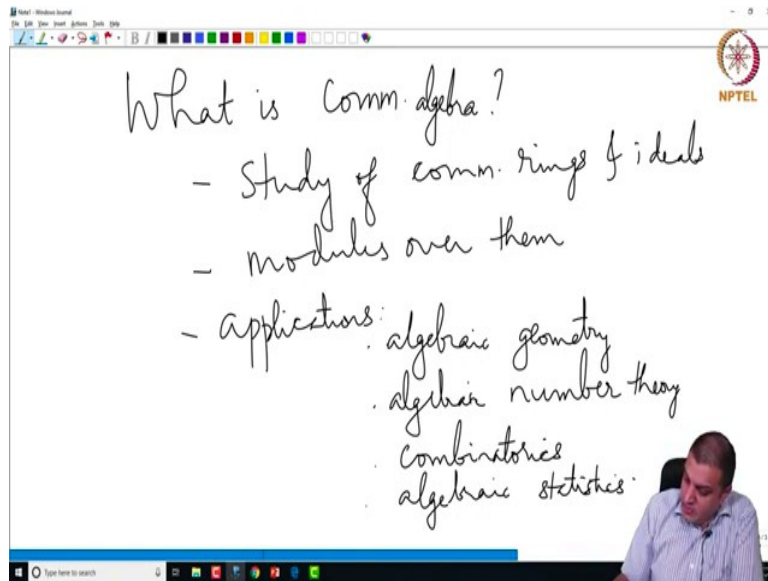
- Website:
 - www.macaulay2.com OR
 - faculty.math.illinois.edu/Macaulay2/.
- Using it on the web:
 - faculty.math.illinois.edu/Macaulay2/TryItOut/ (There are links to multiple sites where one can do calculations.)



So, here are the references. A primary reference will be this book by Cox Little and O'Shea. Then this book by Eisenbud and we will also occasionally refer as a or it might help you occasionally to refer, if this is the first time you are seeing these topics if look at Reid's undergraduate commutative algebra.

And the software that we will use is, as I mentioned earlier Macaulay2, which can be downloaded from it's website www.macaulay2.com or the points much longer URL, which is from the University Illinois. And if you do not want to download it and just want to run some simple programs or simple few commands or for the program, there are places on the web where you can do it directly. And links to multiple such sites are available on this URL: faculty.math.illinois.edu/Macaulay2/TryItOut/

(Refer Slide Time: 04:11)



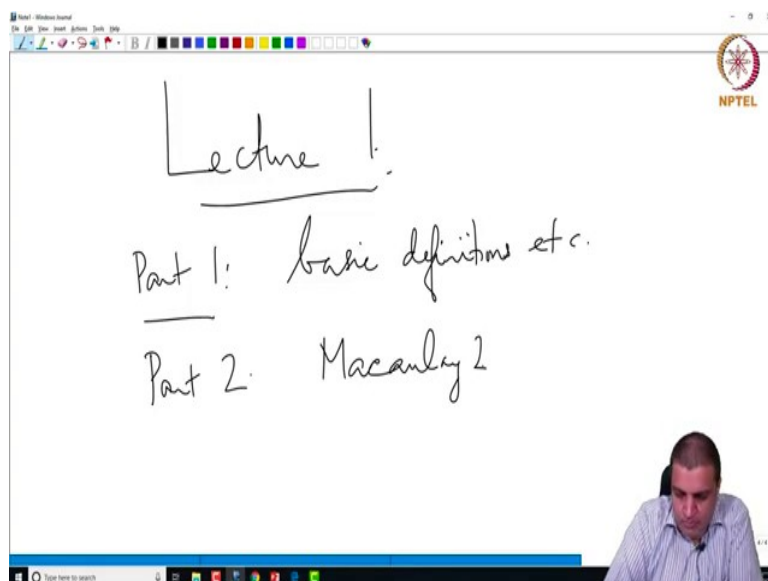
What is comm. algebra?

- Study of comm. rings & ideals
- modules over them
- applications:
 - algebraic geometry
 - algebraic number theory
 - combinatorics
 - algebraic statistics

The image shows a whiteboard with handwritten text. At the top, it says 'What is comm. algebra?'. Below this, there are three bullet points. The first two are 'Study of comm. rings & ideals' and 'modules over them'. The third is 'applications:', followed by a list of four items: 'algebraic geometry', 'algebraic number theory', 'combinatorics', and 'algebraic statistics'. The whiteboard is part of a video lecture, with a small inset of the lecturer in the bottom right corner. The NPTEL logo is visible in the top right corner of the whiteboard area.

So, what is commutative algebra? It is the study of commutative rings and hence also of ideals. It is the study of modules over them and also applications. Traditionally from the 19th century questions in commutative algebra or the importance of looking at some of these questions came from studying questions in algebraic geometry, also algebraic number theory and increasingly to combinatorics and also algebraic statistics. These last two topics are relatively more modern. And applications to the last two topics typically are related to the computational aspects of commutative algebra.

(Refer Slide Time: 06:01)



Lecture!

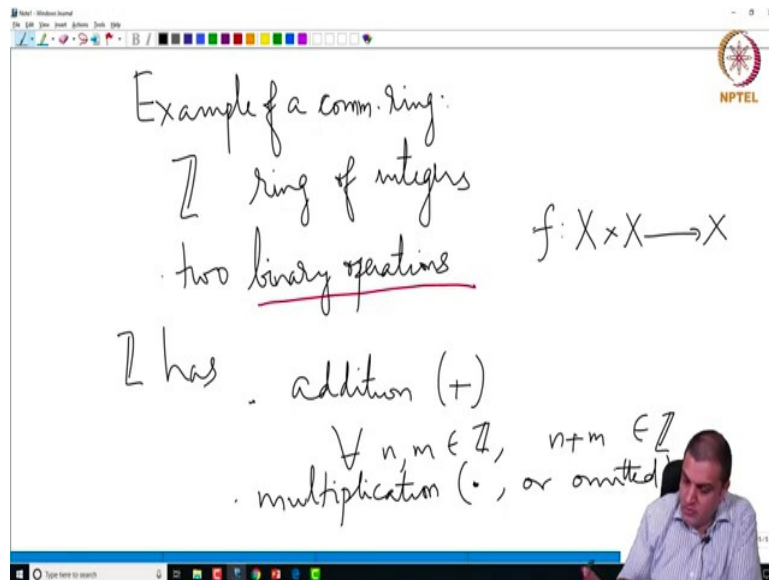
Part 1: basic definitions etc.

Part 2: Macaulay 2

The image shows a whiteboard with handwritten text. At the top, it says 'Lecture!'. Below this, there are two parts. 'Part 1:' is underlined and followed by 'basic definitions etc.'. 'Part 2:' is followed by 'Macaulay 2'. The whiteboard is part of a video lecture, with a small inset of the lecturer in the bottom right corner. The NPTEL logo is visible in the top right corner of the whiteboard area.

So, this is lecture 1. In part 1 of today's lecture, we will just do basic definitions etc. In part 2, we will do this in Macaulay2. However, in future lectures, we may not make a distinction. As we discuss topics, we will also look at some examples in Macaulay2.

(Refer Slide Time: 06:45)



So, what is an example of commutative ring? The most basic example is \mathbb{Z} , the ring of integers. What does that mean? It has two binary operations. So, what is a binary operation?

So, this is a function f from the Cartesian product of a set X with itself to so i.e. $f: X \times X \rightarrow X$. Such a thing is called a binary operation.

And \mathbb{Z} has two binary operations that are of interest to us addition, which we denote by $+$ and what does that mean? For all integers n and m , we can do take the sum $n+m$ that is also an integer. And it also has multiplication, which we denote by \cdot or just omitted.

(Refer Slide Time: 08:34)

Both operations are commutative.

$$\forall n, m \in \mathbb{Z}, \quad n + m = m + n$$

and $nm = mn$

mult. distributes over addition:

$$\forall m, \forall n_1, n_2 \in \mathbb{Z}$$
$$m(n_1 + n_2) = mn_1 + mn_2$$

And now, what are the properties of these operations? Both operations are commutative. If you take the sum $n + m$ or $m + n$ these are the same and the product is the same. And multiplication distributes over addition, in other words for all integers m and for all integers n_1 and n_2 , if you take $m \cdot (n_1 + n_2)$, this is the same as $m \cdot n_1 + m \cdot n_2$. So, this is one few more properties that, we need interest to us in this thing.

(Refer Slide Time: 10:11)

Additive identity, 0

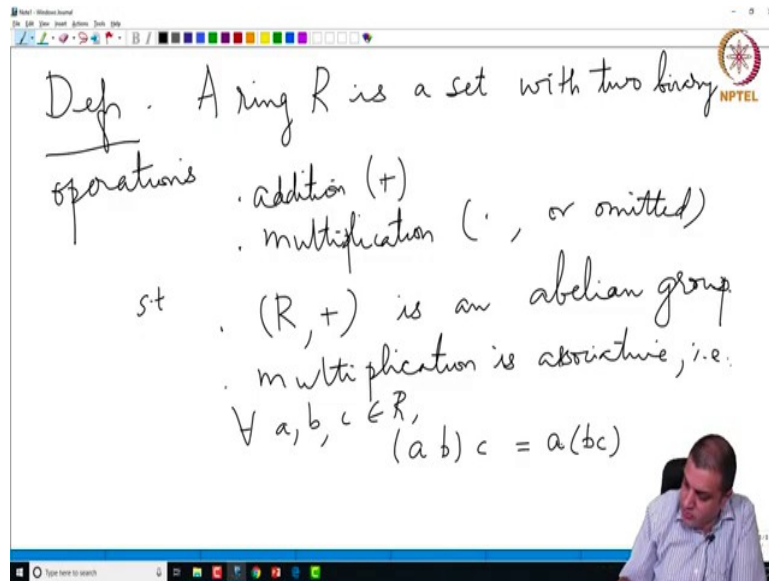
$$m + 0 = 0 + m = m \quad \forall m \in \mathbb{Z}$$

Mult. identity, 1

$$m \cdot 1 = 1 \cdot m = m \quad \forall m \in \mathbb{Z}$$

There is an additive identity 0 and that has a property such $m + 0 = 0 + m = m$. Because, the same is commutative this both of these things are the same value m for every integer m . And, similarly there is a multiplicative identity 1 and that says $n \cdot 1 = 1 \cdot n = n$ for all integers n . So, now, we come to the definition of a ring.

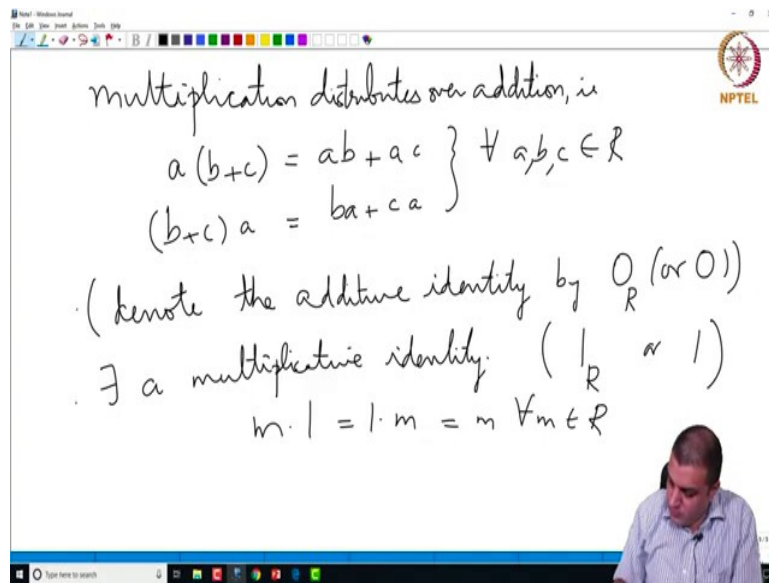
(Refer Slide Time: 11:17)



A ring R is a set with its two binary operations and we call it addition and multiplication. Addition, which we denote by $+$ sign and multiplication, which we denote by \cdot sign or just omit typically, such that under addition so, if you take the set with this binary operation $+$, this is an Abelian group it's an Abelian group.

Multiplication is associative that is if you take three elements in the ring. If you take the products, but remember we can take products of only two things at a time. So, we can do it in two different means we could take the product of a and b first and then multiply by c or we could take the product of b and c first and then multiply by a . And the associativity property says that these two things give the same result.

(Refer Slide Time: 13:50)

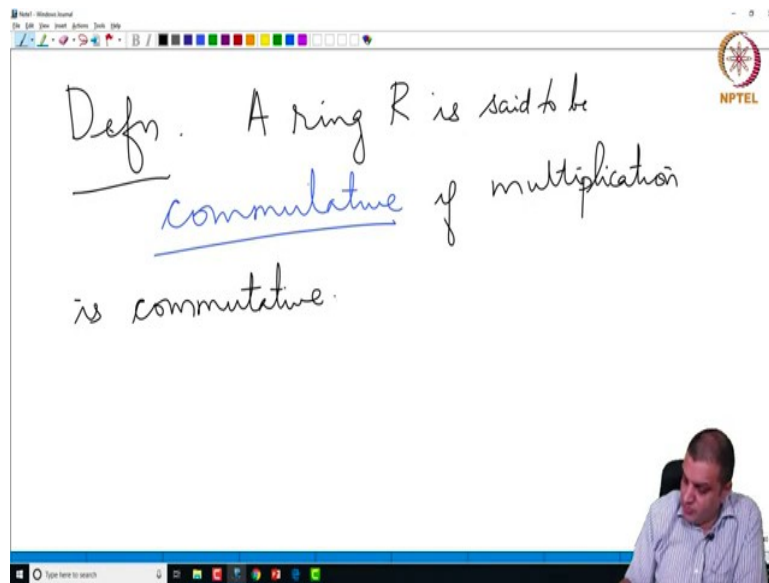


So, multiplication is associative multiplication distributes over addition. That is $a(b+c)=ab+ac$ and we have not said that multiplication is commutative. So, we must also consider $(b+c)a$; these two things may not be the same, but distributive property says that this is same thing as $ba+ca$. And these two statements for every triple $a,b,c \in R$; multiplication distributes over addition.

Then there is addition. We already said that under $+$, the set R is an Abelian group. So, it has an additive identity which we will denote 0_R . So, we denote the additive identity which we already know because it is a group the additive identity by 0_R . And often this it will be clear that which ring is we are talking about.

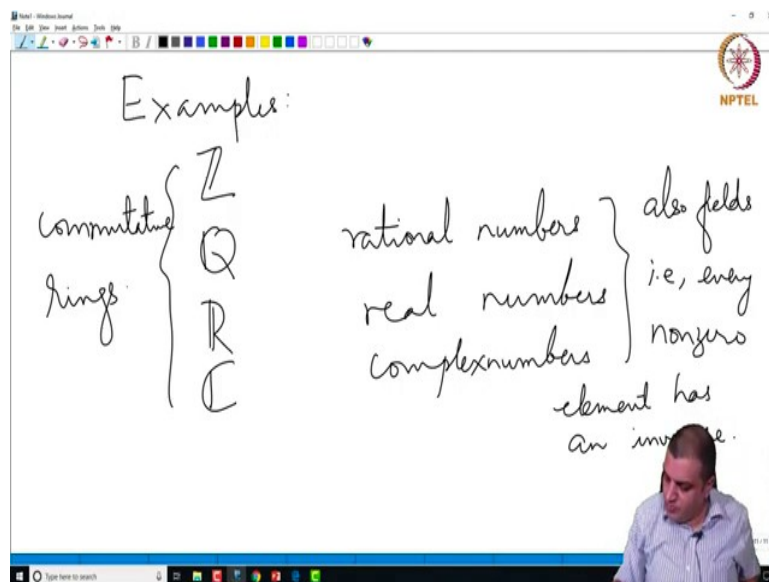
So, we would just write 0. Multiplication distributes over addition. And there exist a multiplicative identity which we denote by 1_R or just 1, there is no confusion as to which ring we are talking about. And so, what does this mean? This means that $m1=1m=m$, for all $m \in R$. So, R is a set with its two such operations and with those properties is called ring.

(Refer Slide Time: 16:29)



Definition: A ring is said to be commutative if multiplication is commutative. The addition is already commutative by definition.

(Refer Slide Time: 17:21)



So, let us look at some examples. The ring of integers that we saw already, then the set of rational numbers, the set of real numbers, the set of complex numbers. These are also commutative rings with the usual addition and multiplication operation.

And in addition to being rings these are also fields, that is every nonzero element has an inverse. Inverse with respect to multiplication or in other words. If we take the nonzero

elements of these sets and take the multiplication operation, they will be a in these cases it will be an Abelian group.

So, these are all commutative rings. So, not all interesting rings are commutative. There are important classes of non commutative rings.

(Refer Slide Time: 19:25)

Matrix rings
 R commutative ring $n \geq 1$ integer
 $M_n(R)$ $n \times n$ matrices with entries in R
- usual matrix addition
- matrix mult.
- $M_n(R)$ ring
 $n \geq 2 \Rightarrow M_n(R)$ is not commutative
add. identity = 0 matrix
mult. identity = I_n

So, let us just to become familiar with that let us just use one example. And this is we will not worry about them in general.

So, let R be a commutative ring and n be a positive integer. By $M_n(R)$, we talk of a set of $n \times n$ matrices with entries in R with usual matrix addition matrix multiplication. With these two things it becomes a ring.

The additive identity is the 0 matrix. And the multiplicative identity is $n \times n$ identity matrix. However, if $n \geq 2$, then this is a non commutative ring. So, now, we look at one more class of rings. After that we look at some examples in Macaulay.

(Refer Slide Time: 21:30)

The image shows a digital whiteboard with handwritten notes in black ink. The text is as follows:

- Polynomial rings.
- R commutative ring
- X_1, \dots, X_n variables
- $R[X_1, \dots, X_n]$ polynomials in X_1, \dots, X_n with coeffs from R
- usual polynomial addition and mult.
- add. identity zero polynomial, mult. id = constant 1.

The whiteboard has a toolbar at the top with various drawing tools and a small NPTEL logo in the top right corner. A small video inset of a man is visible in the bottom right corner of the whiteboard area.

These are called polynomial rings. To be start with a commutative ring and we can take some variables. Let us say X_1, X_2, \dots, X_n variables. And then by $R[X_1, \dots, X_n]$, we mean the set of polynomials in the variables X_1, X_2, \dots, X_n with coefficients from R . And this is a commutative ring with usual polynomial addition and multiplication.

And the identity additive identity is the 0 polynomial. So, this is the additive identity and the multiplicative identity is the constant polynomial 1. And with these things this is a commutative ring.

So, we will stop here now, and we look at a few basic examples in Macaulay.

(Refer Slide Time: 23:25)



Macaulay2, version 1.14
--loading configuration for package "FourTiTwo" from
Macaulay2/init-FourTiTwo.m2
--loading configuration for package "Topcom" from f
Macaulay2/init-Topcom.m2
with packages: ConwayPolynomials, Elimination, Inte
InverseSystems, LLBases, PrimaryDec
TangentCone, Truncations
i1 :



So, when you start up Macaulay, it depends on what you would see in the terminal or the output will depend on how its installed, what operating system, it is etcetera are you running, it on the web or are you running it locally etc.

So, here I just run it on terminal and I just show you the screenshots of what one would see and when I run, it I get a line which says this is version 1 14. And this is the input line; this is the input line the 1 that starts with an I says you are in a input prompt. And it says the first input prompt.

(Refer Slide Time: 24:05)

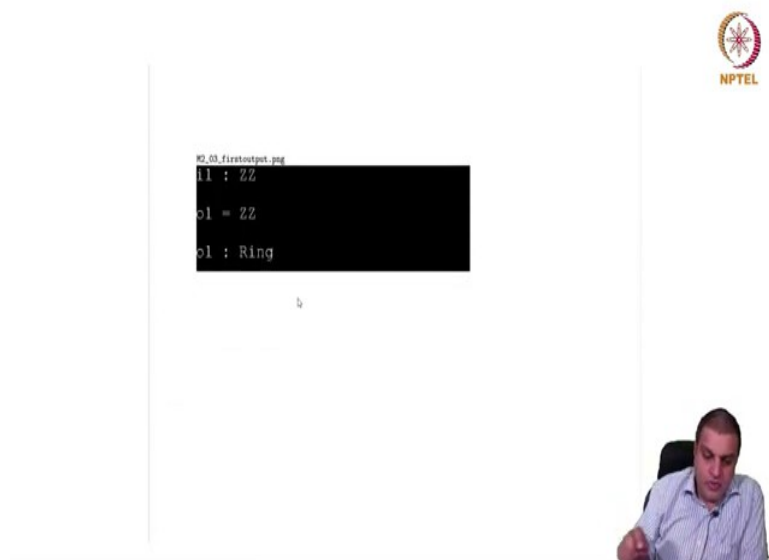


Macaulay2, version 1.14
--loading configuration for package "FourTiTwo" from
Macaulay2/init-FourTiTwo.m2
--loading configuration for package "Topcom" from f
Macaulay2/init-Topcom.m2
with packages: ConwayPolynomials, Elimination, Inte
InverseSystems, LLBases, PrimaryDec
TangentCone, Truncations
i1 : 22



So, now let us give something in the first input prompt. The first ring that we know we type ZZ here and then press enter.



(Refer Slide Time: 24:16)



So, here is the output. It says the first output line. So, input lines are marked with an “i” output lines are marked with an “o” and it says o1 is ZZ that is just acknowledging what it understood. And then it says o1 is a Ring.

So, here this is the class of this object and we do not need to worry about it a lot. But, soon we will have to worry when you look at various functions we will have to see what the type of input is what type of output is and for that at that point. We will have to worry and Macaulay tells at each time whatever, it sees what object and where is it coming from.

(Refer Slide Time: 25:02)



So, now let us do a few more things, let us explore a few more things. So, in the second line of input i , just type 4 and output is 4 itself. And that is because it understands this as an integer. Then I ask what X as a third input line and it says X and its a symbol. That is it does not know where this X came from. Here, it knew it was an integer.

(Refer Slide Time: 25:33)



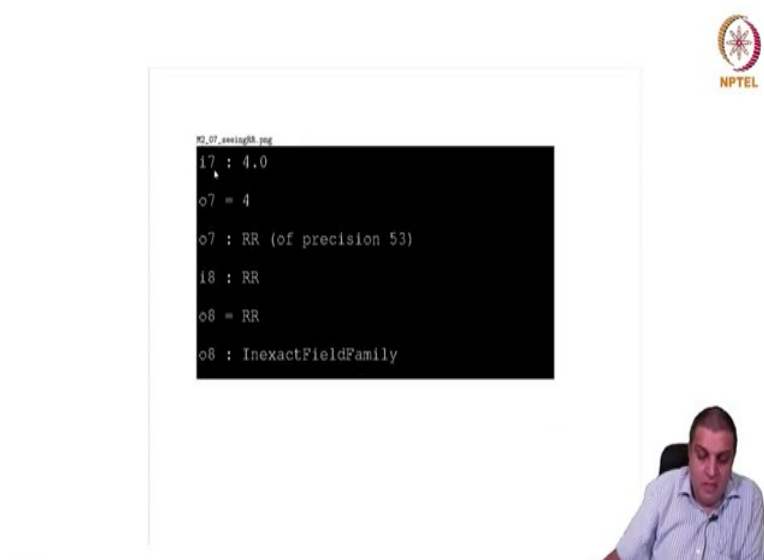
So, now we look at rational. So, I type $4/1$. So Macaulay computes and says it is 4. Now, it says it is QQ that is the set of rationals and this is something that we just saw. So, we asked Macaulay QQ , then it says QQ it is a ring. And this is the notation for the set of rationals.

(Refer Slide Time: 26:09)



So, whenever we are confused we can ask for help. So, now, I type help QQ and one has to present here help QQ. So, then it will just type a lot of help and it will run into multiple line. So, I am not showing what the output is. You can try this yourself. But, it essentially tells you what are the things that, you can work with in Macaulay with QQ. What are things that you can do? Its a long output. So, if I hidden.

(Refer Slide Time: 26:39)



So, now we come to something new i so, real numbers I type 4.0. So, it again says its 4, but its not just integer, it says, it is a real number. *RR* denotes real number. But, this time there is

a difference it says of precision 53 earlier ZZ and QQ did not say that and so, I type *RR* and unlike a ring it now says in exact field family.

And the reason for this is in finite precision, you cannot write all real numbers. So, its a family of depending on the precision that one needs *RR* is implemented as a family of fields. And here the 53 is the default precision and it just shows that.

(Refer Slide Time: 27:27)



```
PS_08_Constant.juy
i9 : pi
o9 = pi
o9 : Constant
i10 : ii
o10 = ii
o10 : Constant
```



Now, I ask what is *pi*? You can say its a constant. Then, I ask what is *ii*? That is also a constant, but what is *ii*? Let us check.

(Refer Slide Time: 27:40)

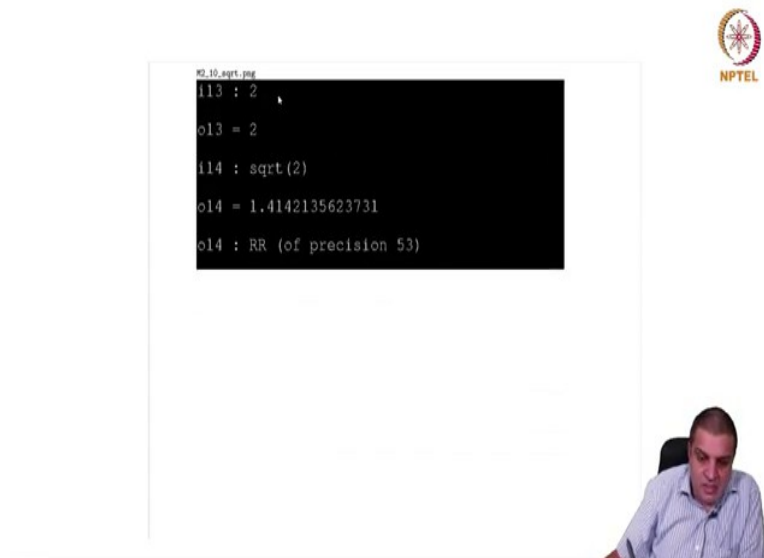


```
PS_09_CC.juy
i11 : ii^2
o11 = -1
o11 : CC (of precision 53)
i12 : CC
o12 = CC
o12 : InexactFieldFamily
```



I ask ii square. So, again each input line it will put something corresponding output and then it will tell you where is output living in. So, ii square is -1 and this -1 is not the integer -1 , it is considered as a complex number. And again same precision for the same reason complex numbers cannot be written in finite precision. And CC again its an inexact field family exactly for the same reason as RR is inexact field family.

(Refer Slide Time: 28:12)



Just to familiarize ourselves with more or usual functions in simple functions, I ask 2, 2 is an integer that is all it says. Then ask square root of 2 this is the square root command `sqrt(2)`, then it outputs a real number and says. Now, that output is in class RR and of precision 53. Default precision of reals is 53.

(Refer Slide Time: 28:36)



```
KG_11_ZZ17.png
i15 : ZZ/17
      ZZ
o15 = --
      17
o15 : QuotientRing
```



Now, I ask $\mathbb{Z}\mathbb{Z} \bmod 17$ we know that this is I mean we expect, the way we would do it on pen and paper. We denote this as the residue class of integers mod 17 i.e. $\frac{\mathbb{Z}}{17\mathbb{Z}}$. So, it says it is a quotient ring. So, I have not yet defined a quotient ring, but it just says here is a ring which is of class quotient ring.

(Refer Slide Time: 29:06)

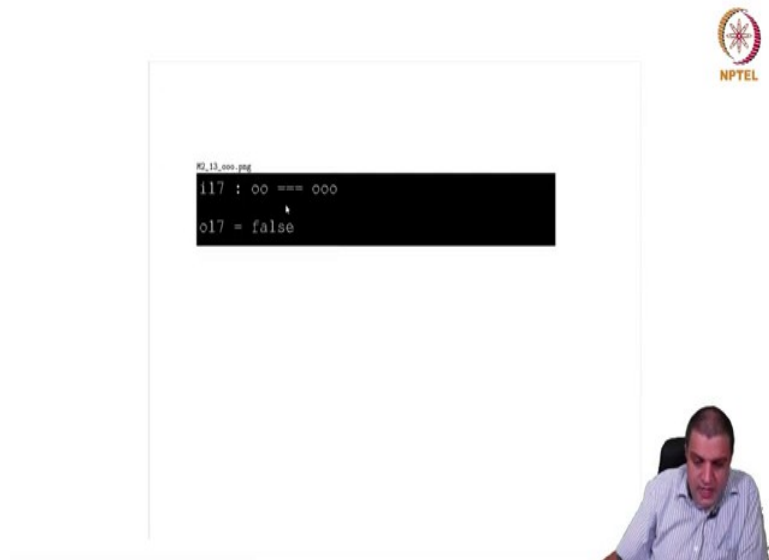


```
KG_12_GF.png
i16 : GF 17
o16 = GF 17
o16 : GaloisField
```



So, we will worry about quotient rings in the next lecture, but we know that this is the same thing as a finite field of 17 elements. And the command to construct a GaloisField a finite field is GF up both uppercase as GF 17 and it says the GaloisField.

(Refer Slide Time: 29:23)



Now, this is where the idea of class comes in. We ask oo I mean 3 equals ooo well. What does this mean? oo means previous output, ooo means the output before that. So, on line 17 oo denotes o16 and ooo denotes o15. And so, its asking whether GF 17 and ZZ mod 17 are the same. And it says its false and that is because, their objects enter one is the quotient ring the other one is a Galois field. So, one has to worry a little bit about these things when you write code.

(Refer Slide Time: 29:58)



```
KG_14_ZZXY.jpg
i10 : R = ZZ[X,Y]
o10 = R
o10 : PolynomialRing
i19 : X
o19 = X
o19 : R
```



So, now I consider a polynomial ring with variables X and Y with integer coefficients that is $R = \mathbb{Z}[X, Y]$. So, now it says R is a polynomial ring. Now, earlier when we asked X its just a symbol, but now if you ask X , it says X is in R , it has found a place where X lives.

(Refer Slide Time: 30:22)



```
KG_15_QQXY.jpg
i20 : S = QQ[X,Y]
o20 = S
o20 : PolynomialRing
i21 : X
o21 = X
o21 : S
```



I now, define a different ring same variables X and Y , but coefficients are from rationals that is $S = \mathbb{Q}[X, Y]$. Again its a polynomial ring. And now ask what X is? It says X is now in S . It just forgotten the R .

(Refer Slide Time: 30:36)



```
KG_16_0406.jpg
i22 : use R
o22 = R
o22 : PolynomialRing
i23 : X
o23 = X
o23 : R
```



How do I go back to R ? I type use R . Now, it comes back to R now if I ask what X is will interpret X is an element of R .

(Refer Slide Time: 30:45)



```
KG_17_0406.jpg
i24 : gens R
o24 = {X, Y}
o24 : List
i25 : vars R
o25 = {X, Y}
o25 : Matrix R <--- R
```



So, so these are some issues that one has to worry about when one writes code otherwise some unexpected results may come. So, how do we know what are the variables in a polynomial ring? So, I asked gens for generators, gens R . It gives X and Y with curly brackets and it says it is list right now.

So, lists are lists of elements with bounded by curly brackets. One can also get this information using a command called `vars R`. But, now it puts the same output, but in a different fashion it writes it as a matrix. So, these distinctions we will not worry about it at this stage, but one must. I mean one should keep in mind that such distinctions do exist.

(Refer Slide Time: 31:33)



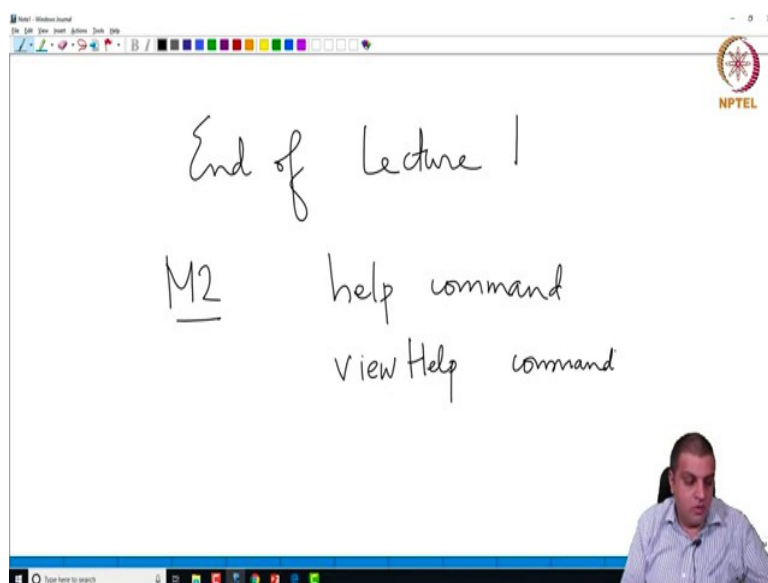
The terminal window displays the following output:

```
i26 : coefficientRing R
o26 = ZZ
o26 : Ring
i27 : coefficientRing S
o27 = QQ
o27 : Ring
```

The lecturer is a man with short dark hair, wearing a light blue button-down shirt, sitting in a chair and looking towards the camera.

We can ask what are the coefficients of these rings come from? So, remember R was a polynomial ring in X and Y over integers, this was a polynomial ring in X and Y were the rationals. So, if you ask coefficient ring take the coefficient ring of R one would get $\mathbb{Z}\mathbb{Z}$ and its a ring and coefficient ring of S is $\mathbb{Q}\mathbb{Q}$ and that is a ring. So, this is the end of the first lecture.

(Refer Slide Time: 32:02)



In the next lecture we will talk about homomorphism's, ideals, quotient rings etcetera. And the and slowly we will try to develop more things about rings, simultaneously understanding how these things are computed in Macaulay, at least how to compute it in Macaulay2, not how what is going on behind it that will take us a few more lectures ok.

So, please redo these commands in the Macaulay code that you saw the lines that start with an i, enter that command you do not need to write the i and up to the colon just the command is enough. And see what the output is and make sure that you understand what the output is saying or at least if you do not understand take a look at their help pages.

So, Macaulay so here after I will start writing M2 to denote Macaulay2, there is a help command to see help in the terminal itself. And sometimes you will run into many pages and you will lose. What we have computed before it?

In that case there is a help command to see it in a browser, if you are working it on a local terminal this would not work, if you are doing on the web, but on the web. So, its viewHelp command H is an uppercase in this thing. So, use these two things to understand at least try to explore what Macaulay2 is ok, see you in the next lecture.