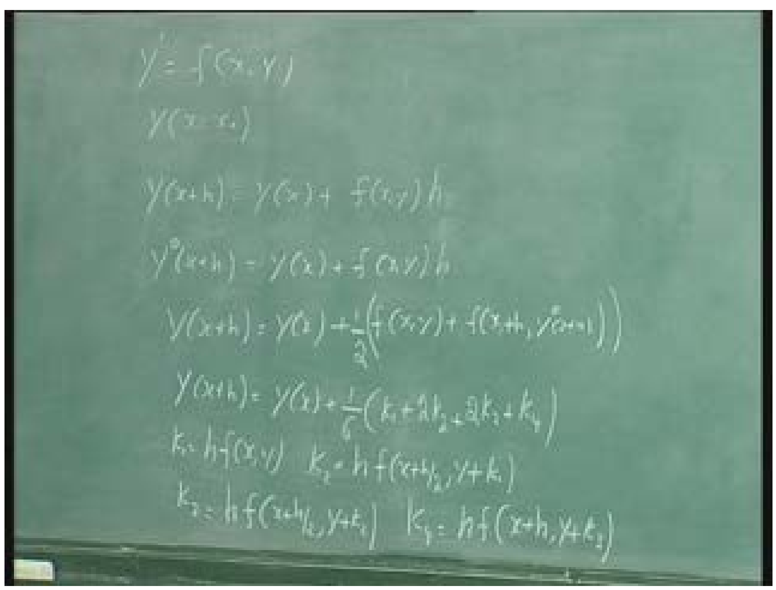
Numerical Methods and Programming
P. B. Sunil Kumar
Department of Physics
Indian Institute of Technology, Madras
Lecture - 33
Adaptive Step Size Runge Kutta Scheme

So far we have been looking at numerical solutions of differential equations of the form y prime equal to f of x, f of x,y first order differential equations with some boundary conditions specified at x equal to x_0 in the beginning of the interval and we looked at 3 different methods actually, 4 different methods, 2 of them being on the Runge Kutta scheme and in these methods, we wrote y at x plus h as y of x plus f into that is y prime evaluated at x, y into h.

So we wrote these methods and so that is 1 method which we said Euler's scheme and then we can go starting from x_0 to the other end of the interval which you are interested in another method which we looked at was to evaluate what is call a predictor which at x plus h from this scheme and then use a corrector step to get the y, the correct value as f of xy plus x, the derivative evaluated at x plus y and the derivative evaluated at x plus h and y_0 at x plus h divided by 2.

(Refer Slide time: 05:16)



So that is another scheme which we looked at and we said that we could do this iteratively and the predictor corrector scheme to get better accuracy and the second order Runge Kutta method which we looked at was exactly this 1. So this is predictor corrector and second orders Runge Kutta in which we evaluate the derivative at 2 points one at the beginning of the interval and 1 at end of the interval with a predicted value of y, predicted value of the function and then taking the mean of that derivative and in the scheme where we looked at the 4th order Runge Kutta.

So then we wrote here y at x plus h as y at x plus 1 by 6 times k_1 plus $2k_2$ plus $2k_3$ plus k_4 and k_1 was f of xy and so k_1 was f xy, f evaluated at x and y and k_2, into h that is what we wrote that as, so $k_1$was h times f of xy and k_2 was h times f evaluated at x plus h by 2 and y plus k_1 and similarly k_3 was h times f evaluated at x plus h by 2 and y plus k_2 and then k_4 was h times, h into function evaluated at x plus h, derivative evaluated at x plus h now and y plus k_3. So this is the scheme which we have learnt so far.

So we want to solve equations of this type first order differential equation with the boundary conditions specified at the beginning that is initial value problem and we looked at the Euler's scheme the predictor corrector scheme and so this is basically the predictor corrector scheme or the second order Runge Kutta scheme and then a 4th order Runge Kutta scheme. So this is what we have so far looked at, now the question would be that if I want to solve differential equations of higher order, higher order differential equations and then what would I do. So how do I proceed in the case if I have to solve a higher order differential equation that is if I had to take differential equations of the form, let us say y double prime is equal to some function of xy and y prime.

So that is now right hand side is a function of xy and the first derivative and this is the second derivative y double prime is second let us say this is a equation of order 2. So let us say I want to evaluate I want to solve an equation of this form between some interval x going from x_0 to x_1 again we are only interested in the initial value problems that is we could say that y is specified at x equal to x_0 and y prime is specified at x equal to x_0.

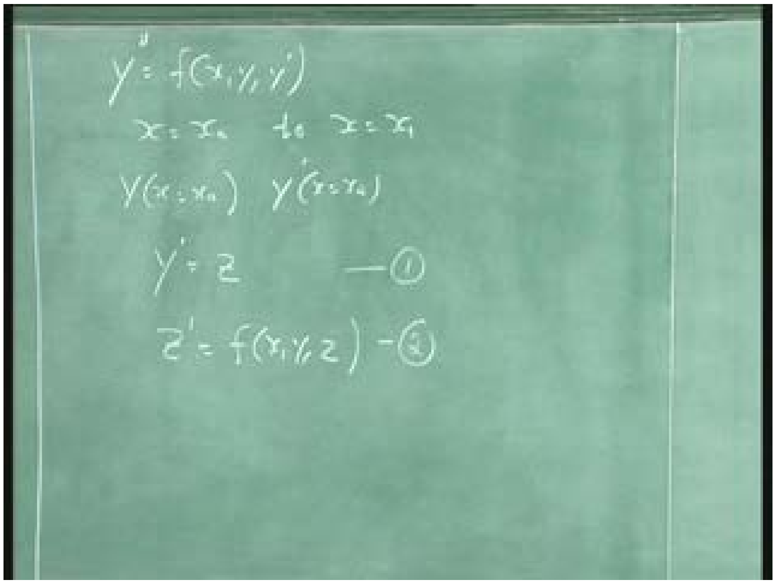
So now this is a second order differential equation, so we need 2 boundary conditions and both boundary conditions are specified at the initial value that is called an initial value problem. Of course, you could have problems in which 1 condition is specified at the initial case x equal to x_0 and the other 1 is specified at the other boundary.

So then that will become a boundary value problem, we can use the method which we learned here to solve problems of that type we will discuss that at the end of this section. Okay so for the time being let us concentrated on this issue that is how to solve an equation of this form with the boundary condition specified at the initial value that is second order differential equation with boundary condition specified at the initial value using 1 of those schemes which we have learnt.

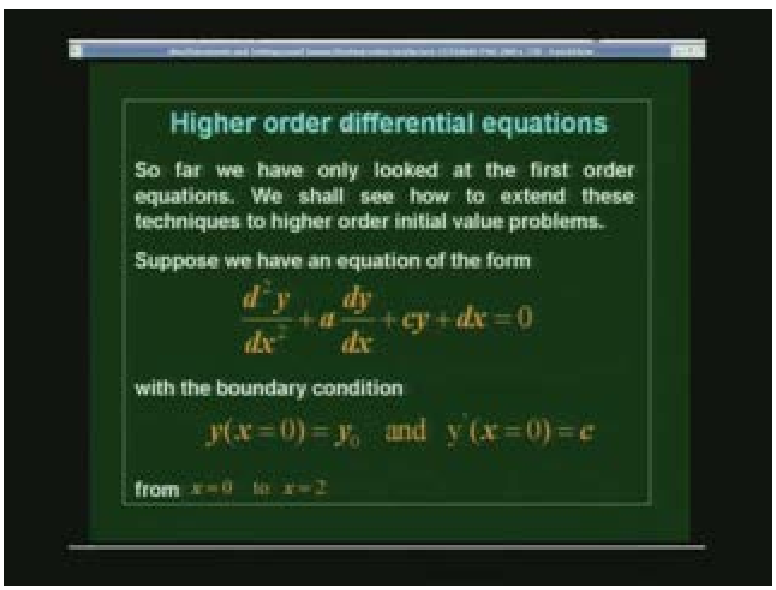
So we would use a Runge Kutta scheme for solving this equation, so what we do is we split this into 2 first order equations that is we would write y prime is equal to z and then say y double prime. So now y double prime that is z prime, so that is z prime is equal to f of x y z. So that is the that should be our equation. So now we have 2 equations instead of 1 second order equation we have 2 first order equations and now we have boundary conditions for this as y at x equal to x_0 and then we have boundary condition for this which is z at x equal to x_0.

So we have boundary condition each 1 boundary condition for each of them and then you can solve these equations. Okay now that is the scheme which you would be following here.

(Refer Slide time: 08:50)



(Refer Slide time: 08:52)



So I will summarize that here we want to let us say for example, here is an example of 1 differential equation of that form. So you have a second order differential equation, so it is d square y by dx square plus a dy by dx plus cy plus d into x is equal to 0. So now we have a differential equation of this form.

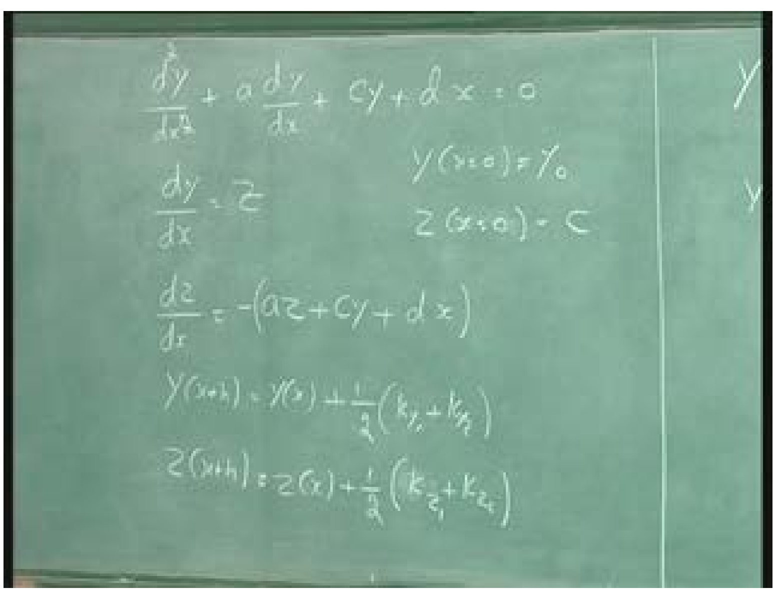
So this is a second order differential equation and then we want to solve this with a boundary condition that y at x equal to x_0 is y_0 and y prime at x equal to, that is first derivative dy by dx at x equal to x_0 is c. Now we have some boundary conditions, we want to solve it from 0 to, x equal to 0 to x equal to 2 for example some x equal to some x value so another value at which is given as 2 in this particular example. So how do we go about solving this, so we said that we can split this into 2 first order equations these 2 first order equations are now given by this dy by dx is equal to z and dz by dx is now to become az plus cy plus xd, let me write that is a minus sign probably there.

So we will write this form, so this equation our equation was d square y by dx square plus we had a dy by dx plus cy plus d into x is equal to 0. So we will write this as, now we are going to write this dy by dx that is y prime we are going to write that as z and we are going to say we are going to say dz by dx is equal to minus a. So that is it, z right minus of az plus cy plus d into x now, we have boundary conditions given as y at x equal to 0 is equal to y_0 and y prime that is z at x equal to 0 as, okay.

Now that is our boundary conditions, now with this, now how do we solve this we would write say y at x plus h right you will write that as, we have done y at x plus we want to solve this z is the right hand side. So I would write this as 1 over 6 times k, let do the second order to begin with we will then solve this for 4th order also. So I will write it as k_{z1} plus k_{z2} by 2. So let us do the second order first so now that is what you would do.

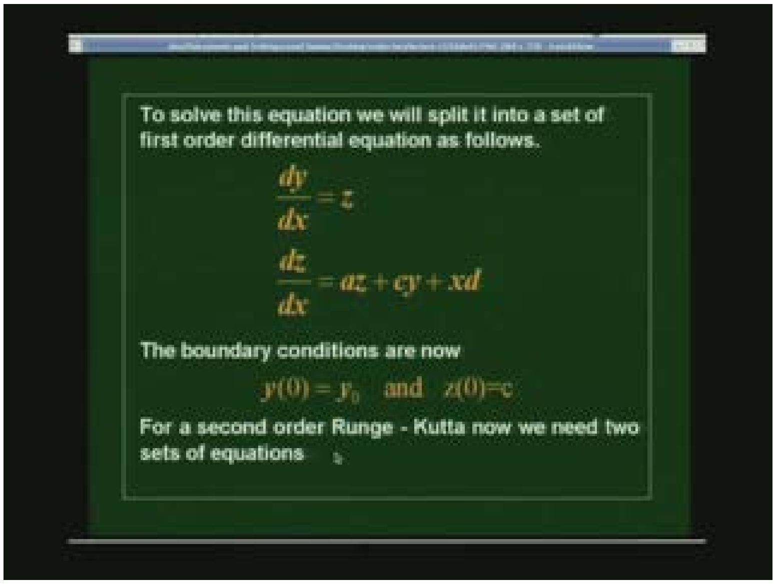
So that is $k_{z1}$and k_{z2} that is what you would write and then similarly, I would write z at x plus h will now be equal to z at x plus half times a into now again. So we would write here I would write it as k_{y1} plus a mistake here k_{y1} and $k_{y2}$and k_{z1} you would write use notation k_y for the y thing and k_z for the z_1, k_{z1} plus k_{z2} by 2.

(Refer Slide time: 13:01)

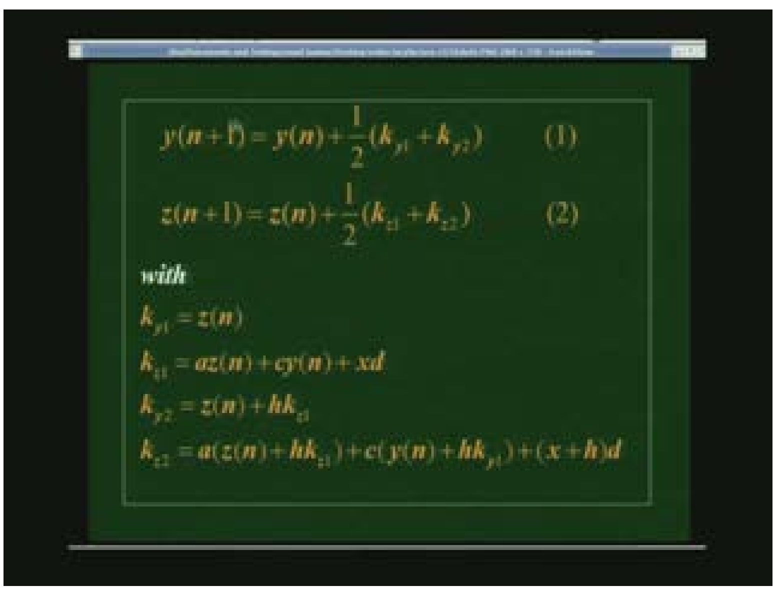


Okay that is what you would write so now this is the 2 equations which we have to solve so now the point is to find out what is this k_{y1} k_{y2} k_{z1} k_{z2} is, that is what we are going to now write down so let us summarize this once again.

(Refer Slide time: 13:14)



(Refer Slide time: 13:15)



So I want to solve this equation, these 2 first order equations. Okay that is actually a minus sign there using a second order Runge Kutta scheme and I would just write it as y at the next point as y at the first point plus this k_{y1} plus k_{y2} and k_{z1} plus k_{z2} for this z_1y and z would go by these 2 quantities. So y will proceed using k_{y1} and z will proceed using k_{y1} and k_{y2} and z would proceed using k_{z1} and k_{z2} starting from the initial value we would compute the next values using this second order Runge Kutta or 1 level predictor corrector scheme. So now the question is what is this k_{y1} would be and how do we write that k_{y1} as so that is your, so remember z next step is z plus k_z and the y next step is y plus k_y.

So now let us compare this with the first order equation which we wrote. So when you write dy by dx as f of x, y. Our k_1 was f of x, y and k_2 was f at x plus h and y plus k_1 that is what we had written as k_2, so now we do exactly the same but now there are 2 such equations. So we have to write now $k_{y1}$and k_{y2} and k_{z1} and k_{z2} etcetera that is all we had to do here. Okay k_{y1} as f at x, y if it is in y this is in y and we say k_{y2} is now f at in general it can be a function of xy and z. So we will say x plus h right y plus k_{y1} and you can have z plus k_{z1}.

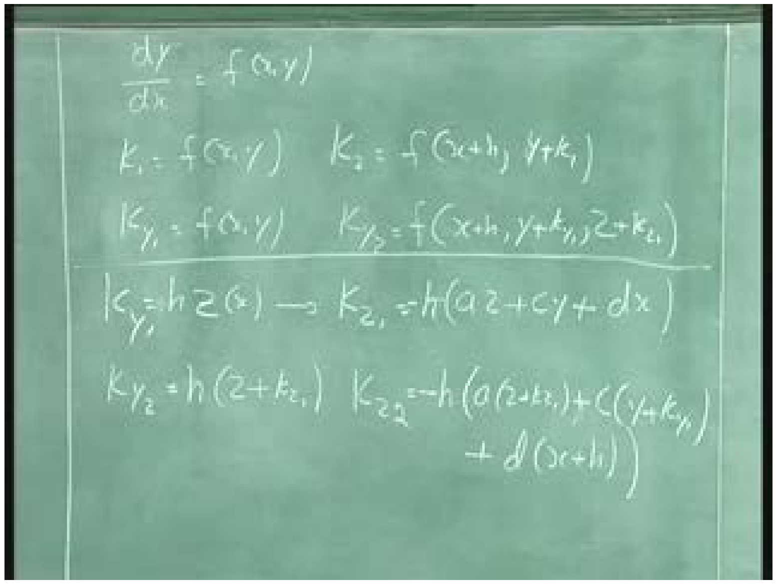
Okay that is the only difference so you see this second equation here can be a function of xy and z, so the xy and z okay so to evaluate k_y. We were to first get k_{z1}, so we cannot evaluate this in 1 shot. So we have to do that in the series that is the only thing which we have to be careful about so when we compute this 1 here, so let me compute this for this so we want to do this first we have to get k_{z1}. So we will write it in this form.

So we get k_{z1}, so our idea is to use this we want to evaluate z at x plus h and y at x plus h so to do that we need k_{z1} k_{z2} k_{y1} k_{y2}. So the way we are going to go about this is first to get k_{z1}, so that is in this case it is just z that is just that the function, this is the first equation dy by dx and that is equation 1 and this is equation 2. So let me write that carefully, so I will say k_{y1} that is the right hand side of dy by dx is equal to that that is z at x and so that is first point x is equal to 0and then I would do k_{z1}. So then I would do k_{z1} which is this quantity which will be minus az plus cy plus dx that will be the second quantity

So the way I go to do is that. So first I compute k_{y1} and then k_{z1} and then I will do k_{y2} this I can compute because I am going to say it is so we are going to use a h here for everything. So minus h times that so minus plus h times z plus k_{z1} that is what I am going to use to compute k_{y2}. So we can do k_{y1} and then we have to do k_{z1} but before we do k_{y2} we have to do k_{z1} because we need k_{z1} there and then I can do k_{z2} that is equal to minus h times a plus k_{z1} plus into and the c times y plus, minus h into a times z plus k_{z1} plus c into y plus k_{y1} plus d into x plus h that is our k_{z2}.

So now we have k_{y1} k_{z1} k_{y2} and k_{z2}, so remember that before we compute k_{y2} we have to compute k_{z1} and before we compute k_{z2} we need k_{y1}. So we have to always do the 1 level thing and then go to the next level in this when you spread the equations into many first order equations, so you could do this for any number um any order that is if you could do a 4th order equation and then you will have 4 such first order equations and then you will have let us say k_1, k_2, k_3, k_4 for each of in this second order case you will have k_1, k_2, k_3, $k_1$and k_2 for each of these equations. So care should be taken in just in do the and when you program it the correct order it has to be computed in the correct order. That is the most important thing and that is what 1 can make mistake in this particular case.

(Refer Slide time: 19:02)



So let me repeat this here. Okay so you had to solve these equations, that is as y_n plus 1is y_n plus k_{y1} by 2 plus k_{y2} second order and similarly for z. So when you compute as you can see first you compute k_{y1} and k_{z1} and then you compute k_{y2} and k_{z2} and then you will proceed in this fashion. So this h cane be written either along with the k or with the on the right hand side of this function. Okay since, we had not included h here, so we have to have h on these equations.

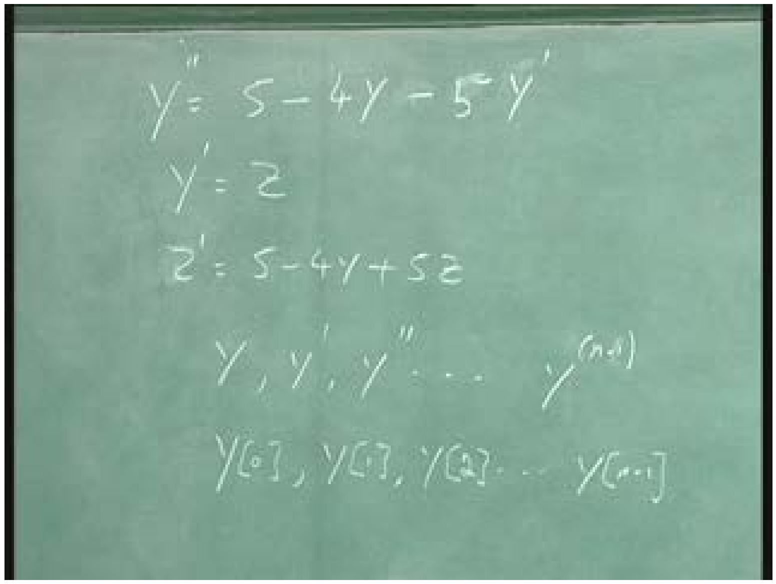
So that is the general procedure and you can see that we can do the same thing for the 4th order case also, we will see that in a program this is the program, this program looks a little complicated but in fact this is extremely similar to what we have done in the earlier case that is for the 4th order Runge Kutta but now we have 2 equations.

So let us straight away go to the function, so the function which we are trying to solve is the following I split that function into 2 first order equations. So let me write the second order equation and then split that into 2 first order equations, so the quantity which we want to solve is the following it is y double prime is equal to we are going to write 5 minus 4y, 5 minus 4y plus 5 times y prime. So that is the equation which we are going to solve so 5 minus 4y minus 5y prime that is the equation.

We are going to write this as 2 things we are going to write y prime is equal to z and z prime equal to 5 minus 4y plus 5z. So in the program in general we could have an nth order differential equation and n such first order equations. So since our aim here is to solve these things numerically we should also see how we develop an algorithm for this.
So one thing which we should we will do is that when you have an nth order equation like this each the solutions will now be both nth order equation like this and we will have solution of y and y prime and y double prime all the way to y_n right that is n minus 1.

So that will be our solution set we will have all of them as a function of x, so we call them as y_0 we put them in an array and this is called y_0 and y_1, y_2 etcetera to y_n minus 1that is what we will do in the program.

(Refer Slide time: 22:14)



So remember when you have a higher order equation, so I write each of those solutions as the elements of an array. So now this is the y, y_0 is y, y_1 is the first derivative y_2 is the second derivative etcetera. So our equation in that language will transfer this z is now y_1 so now my equation is actually y prime equal to y of 1 and y_1 prime is equal to 5 minus 4 y_0 plus 5y minus actually minus $5y_1$.

So that is the way you would write. So y_0 is y and y_1 is the first derivative of y that is what you will write here so that is the way I have written here so this is the first equation this is the first derivative so you have 2 derivatives now that is the first derivative is just y_1 and the second derivative that is the derivative of y_1 is given by derivative of derivative of y_1 that is the second derivative is given by 5minus 4y minus $5y_1$.

So that is what the equation is we are going to solve. So now we have the other boundary conditions, the boundary conditions are specified in the main program that is my main program here. In the main program, we have the upper limit and the lower limit, so I am going to solve this, the limit x equal to 0 to x equal to 2. So now what we are going to solve this one, so we are going to solve this 1 from x equal to 0 to x equal to 2. So that is what we are going to do and then now we have as I said y is now I got 2.

So x and y are pointers and this is now, I have 2 elements in this array so I am getting a 2 here, so there are 2 elements here in this array y_0 and y_1 because my order is 2. So put that m as 2 that is my order, so this specifies my order and n would be the number of points I would use to solve but I am going to do this in this particular program using adaptive step size, so this has everything now.

So I can change the order here and then this program would work for any order of the differential equation provided you give all the boundary conditions at the initial position and it will run for a using adaptive step size. Okay let us look at that, so now we have the boundary conditions specified here both the boundary condition is y_0 y_1 both are 0, that is so we have xy at this point y is 0 and this point z is 0 at x equal to 0, y is 0, z is 0 that is what we have, that is that translates in this language to at x equal to 0, y of 0 equal to 0,y of 1equal to 0. Okay so that is our boundary condition.

So now given that boundary condition I can start from x equal to the lower limit and go all the way to the upper limit. So I start with x equal to the lower limit, so x is a pointer here which gives you that so now I call Runge Kutta scheme of order 4 and I pass the order of the equation right this is we saw in the last class, that this is the step size which we suggest. So h is the step size which we suggest which actually it asks for here.

So the program asks you to enter the station for the step size and then it reads the step size from the screen and it will modify the step size by using an adaptive step size algorithm. So here is the step size and here is the pointer to the function which contains all my derivatives, so now this order will be passed on to will be important in dimensioning the derivatives, how many derivatives are these.

So if it is 2m is 2 there are 2 derivatives and x and y are the values x is the value at which we want to now do the calculation. So we go from x to x plus h and y is the solution set which will be returned and h_n would be the distance to the next step, the adaptive step size would tell you what the next step should be used as that is what we will put that back and go back, we will put that as h here and goes back here and goes to the next x value.

So this program would return to us the next x step and so if you give x_0 y_0 to start with here we give a x_0 y_0 and it returns the solutions at x plus h and the y values at x plus h that is y and y_1 y_0 and $y_1$that is the first, the function and its first derivative because the order is 2. So that is what this program would do.

Okay that is the main function now this just calls this program. Now in this program, now we have to have the y's which is again the same dimension as the y's and then now we have the derivative here now the derivative d y d x it is also now a which I call the dy dx which is a derivative this also now is an array of dimension 2 that is 1 thing which you have to remember. So now this dy dx is a derivative is a array of dimension 2 that is the derivative of the this equation and the derivative of that.

So now this derivative is called dy dx 0 and this derivative is called dy dx1.so these are the 2 equations which we are going to solve so this is this left hand side is called dy dx0 and this is called dy dx1 that is what we are going to use.

So basically d y dx0 will appear in all k's k_y once that is the first derivative so all the ky_1 is that will enter and the k_{z1} it will enter as dy dx 2, dy dx1. So now let us see that so now again so we are going to use an adaptive step size as I said, so here we will now start with these values, so whatever x value is given to this program. So let me write this scheme here. So remember, so we want to start from x equal to 0 so we start from x equal to 0 and we are going to give a so in the main program.

So the main program what it does is, it takes some main program just takes ax and y value it takes ax value and y at x value and y prime at that x value that is what this program is going to do and it is going to call from this main program and it is going to call this Runge Kutta 4 that is why I call rk_4, the Runge Kutta 4 and then so this program would calculate y at x plus h and y prime at x plus h.

So it would calculate that given this so now to determine this h, so we also pass an h to this program as a guess value but this program would actually call, it will actually look for, it will return to this 1 these values y at x plus h, y prime at x plus h and a h next, so h next which I call h_n all these things are returned to this and actually x plus h.

So that will be returned to this and now what we do here is to now replace these quantities which is going to be passed on to this as these quantities that is y at x plus h and y prime at x plus h and x by x plus h next and then it goes back here as the initial value and then it will compute the next step and comes back here again. Okay that is what we wanted to do, I hope that is clear. So that is what is done here, so now we have we have got this y_0 y_1, so remember each time this program is called that is each time this rk_4 Runge Kutta 4is called it gets a new x_0 and y_0 value. So that is from the main program each time this calls this function it gets a new value of x and y.

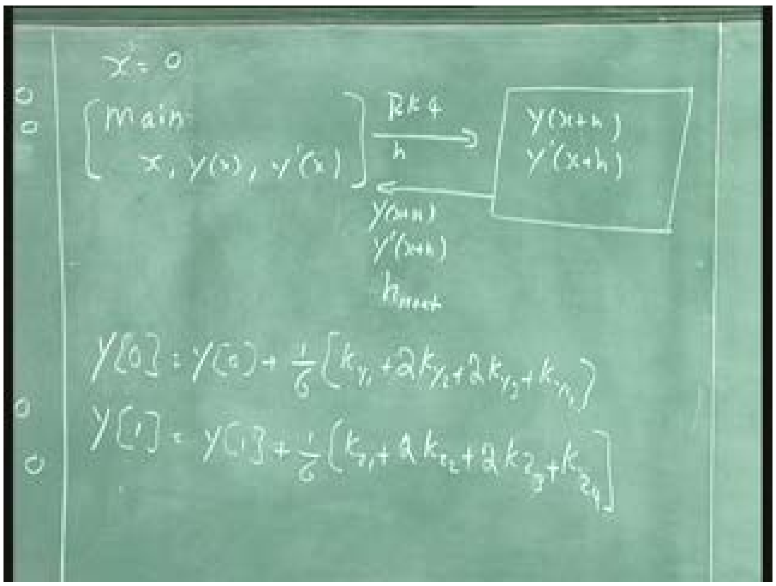
So that is what it is going to use for the next time step so this gets x_0 y_0 as the value and it just simply writes it on to a file and it goes back and calls it again. So every time it is called it is going to call with the new x_0 y, that is x plus h and y at x plus h and then x plus 2h and y_0 and $y_1$at x plus 2h etcetera. So for this program it is always getting y_0 and y_1, so for this guy which does the rk_4 it is always getting these values x and y prime at x what is that x keeps changing as we make each calls. So that is the part here and then we store that thing, so and then we will as we have d1 in the first order case the first order differential equation now the only change here is that we have instead of y alone now we have y_0 and y_1.

So now we have the initial values given as xx and x that is xx is the x value at which we are going to evaluate this and then y_0 at that point which I call y_{01} because $y_{01}$is stored as y_{01} and y_{11} is stored as y_1 and then I will go here and then I compute called this a function called derives remember this is exactly the same as the first order differential equation case only thing is now this derives which is the function which we looked at now returns 2 derivatives dy dx_0 and dy dx 1 and we have passed 2y values y_0 and y_1 that is the point. So now and then using that y_0 and $y_1$and dy dx 0 and d y dx 1 we compute the new values of y_0 and y_1.

Okay so now this idea is to do now we are doing a 4th order Runge Kutta and we want to compute y of 0 as y of 0 plus 1 by 6 times k_{y1} plus k_{y2} plus k_{y3} plus $2k_{y1}$ plus $2k_2$ plus $2k_{y3}$ plus k_{y4}. Okay that is what we are going to compute, so we do each step, so one we compute k_{y1} and I will update my y_0 as y_0 plus k_{y1} by 6and similarly y of 1, I will write y of 1 as 1 by 6 times k_{z1} plus 2 k_{z2} plus 2 k_{z3} plus k_{z4} okay.

So now this is the dy dx 2 these are dy dx 1, 0s and these are dy dx 0 and these are dy dx 1. So again when I compute this I will compute k_{y1} and k_{z1} that is dy dx 0 and dy dx 1. I will update this, I will do this step first and then I will do this step then I go around and compute k_{y2} and k z_2 so every time I compute this thing I update my y by adding that part into the y_0, so that is what this program is doing.

(Refer Slide time: 34:36)



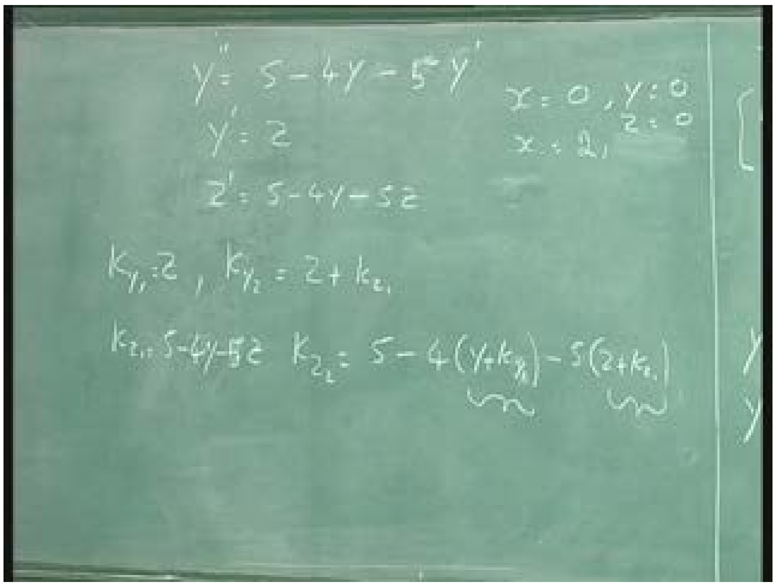
So, first step is to compute k_{y1} and k_{z1} that is returned as dy dx 0 and dy dx1 in this thing because it is in an array here. So the advantage of writing this in the array form is that you can um extend this program to any order by just changing the order m by setting m equal to 2, 3, 4 etcetera.

So then this array dimension would automatically change because we have dimension data is m plus 2 here. So we get array dimension automatically changed that is the first step and then we go from x going from x plus h by 2 that is the next step and then we will compute now k_{y2} and k_{z2} that is again the derivatives at that values. So that is what we are going to compute there.

So now we are updating the y values, so now we want to compute k_{z2} and k, remember if I want to compute k $_{z1}$ in this program this is the equation which you are going to solve right. So, $k_{y1}$is z plus k_{y2} is z plus k_{z2} right. So we will write that again here. So then I want to compute this, so what I need, so I computed first k_{y1} and k_{z1} and I have added that into this part, that is finished.

So now I want to compute the next step that is k_{z1} k_{y2} and k_{z2}. So remember k_z, k_{y2} is y_2 is actually in this case z plus k_{z1} that is what k_{y2} is, okay this is k_{y2} what is coming as y here is the k_{y2} . So that is k_{y2} is that dy by dx equal to z. So this equation has the k_y and this solution of this gives us the y is a solution of this gives us the y_1 that is the z. So this has the k_{z1} k_{z2} etcetera and this will be k_{y1} k_{y2} right hand side of this is z. So k_{y2} is z, k_{y1} was z and k_{y2} is z plus k_{z1} and k_{z1} was 5 minus 4y minus 5z. Okay 5z and now k_{z2} would be then 5 minus 4y plus k_{y1} right. So that is here a minus 5 into z plus k_{z1}. So in this program what I do is I pass this new y value and then new z value into the function and it will return the right hand side for both k_{y2} and k_{z2}. So that is what it would return.

(Refer Slide time: 37:38)



So I just computed that y value y_0 and this is y and this is z_{y1} is z and y_0 is y remember that. So now when it is updated as derivative by 2 and this divided by 2 in the case z_2 there is half here. So that is what we are doing. So we compute this at the k_{z2} and by passing this function this as a new y value into the function which returns the derivative. So I have a function which simply returns the right hand side of these 2 equations to that equation now I pass z as that and y as this and it will return the right hand of these 2 as dy dx 0 and dy dx 1.

So that is what is just done here. So I updated my y_0 and y_1, okay and yy_0 and yy_1 and then I pass that into this function and then we will return the dy dx 0 and dy dx 1 and then that will give me the derivative and I use now this equation this equation and I add this term now to the y value, that is k_{y2} by 3 and k_{z2} by 3 that is what I am doing here, that is what is done here.

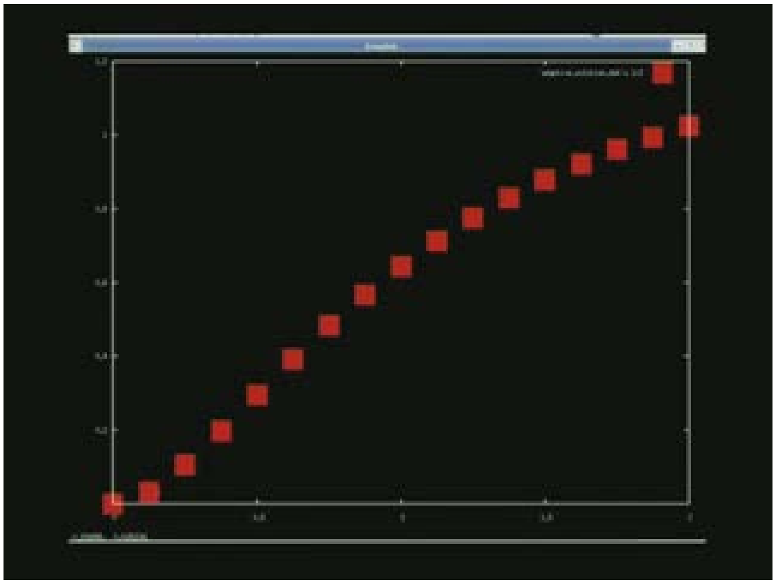
So I will update that thing here and then I do that similarly for k_{y3} now this step is straight forward, so now this is same as this now I do for k_{y3} again I update my yy_0 and yy_1 call the derives to get the derivative and update my y_0 and y_1 and similarly, for k_4 and then it is the first loop when I do this, I store them that solutions I got as y_{n0} and y n_1. So that is my predictor in some sense it is my predictor and then this predicted value with step h and then I reduce by step to h equal to h_2 here I reduce by step to h equal to h by 2 and I do the same thing now for 2 loops.

That is h by 2 step and x plus h by 2 and x plus h and I get the new y values as we saw in the last class for the case of a first order equation same thing we repeat for the second order equation for with y_0 and y_1 and in the earlier case it was only y_0 that is the only difference and then you compare the error which we get in both cases now there are 2 functions so y_{n0} minus y_0 by y_0. So remember, yn_0 is what is the y value obtained at x plus h using 1 step that is go from x to x plus h in 1 step and y_0 is a y value obtained by going from x to x plus h in 2 steps, x plus h by 2 and then x plus h, now these 2 step values we compare these 2 values that is yn_0 minus y of 0 divided by y of 0 for the absolute value of that.

So then similarly, for y_1, so and then I add this construct my error function there, I construct my error function here in this way and then I look at my error is less than my tolerance or not if it is less than the tolerance I increase the step by 2 times the h value which I use, if it is greater than the tolerance I step the, I reduce the step size further to h by 2 and I restart the whole program whole thing again the whole calculation again.

So that is something which we saw in the last class and then I returned the y values and the x values, the x next values to the main program and the main program will call it with again with the new x values and new y values and the process would repeat. So let us and so then in the end we are solving an equation of this form remember that is d square y by d x square is equal to 5 minus 4 y minus 5 dy by dx. So let us look at this we will just run this program and see there so that is, so now we need to pass the step size the order of the equation is 2, the second order equation and the upper limit is 2 and the lower limit is 0 it is solving from 0 to 2.

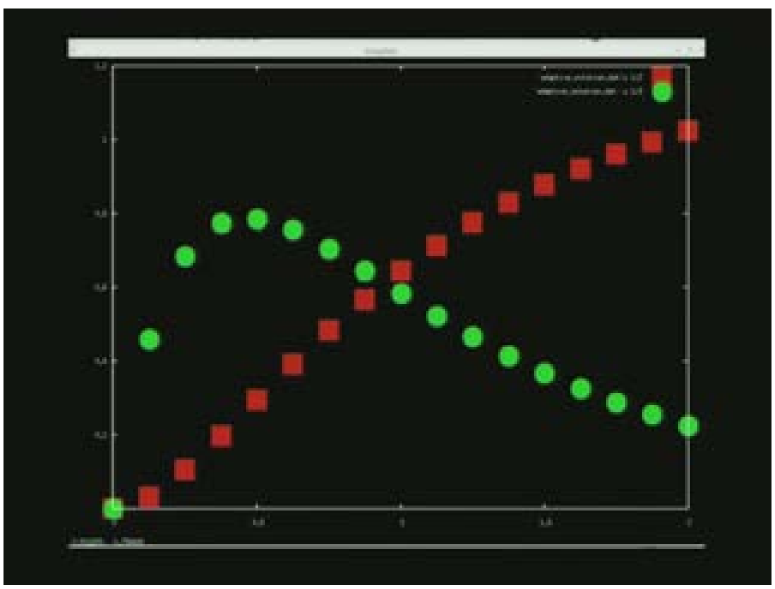
(Refer Slide time: 43:02)



So I would just enter a step size let us say “point 5” and it will solve that now we just brought that function so we get the data file we get is actually adaptive solution dot dat. So we will plot that, so we have now in the solution set we have now this I should just show you that solution how it is written, now the solution is written in this form in the main program I write the solution as x that is the x value and the y value and its derivative. So if I use 1 colon 2, 1 and 2 then it is x versus y and if I use 1 and 3 then x versus y prime that is the derivative of y so the complete solution as I said the second order equation the complete solution means actually obtaining both the function and y as the function of x and it is derivative as a function of x so that is what, so now here is the solution of the equation.

So now this is x versus y I am plotting x versus y and this is the y value as the function of x, you can also plot the x you can also plot the next part that is the derivative. So that will be using 1 colon 3 that is x and it is derivative. So as I said the solution is written as first column is x, second column is y and the third column is the y prime that is the derivative, okay now that is the derivative.

(Refer Slide time: 43:36)



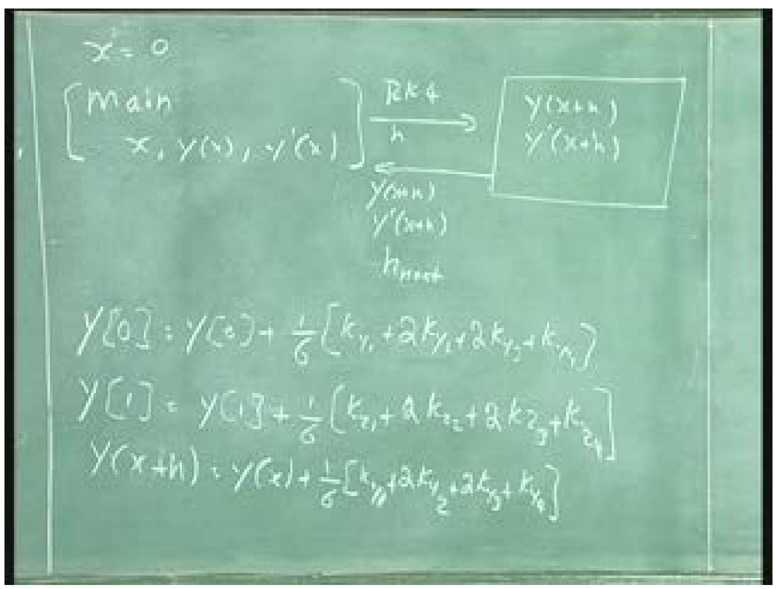
So you can have the solution the y the solution 2 function that is the y value and its derivative as you can see the derivative increases here and it decreases here because this function is saturating to some value here. Okay so now that is the solution of the equation of this form that is y double prime equal to 5 minus 4y minus 5y prime it is a second order equation with 2 boundary conditions specified at the initial value x equal to x_0.

So now we end the discussion on ordinary differential equations here with just 1 more point that is, so far we have only seen how to solve the first order equations the initial value problems we saw how to solve order nth order equations or nth order ordinary differential equations as an initial value problem in the case where some of the boundary conditions are specified at 1 point and the other boundary conditions are specified other m minus n boundary conditions are specified at the other end for example, in this case you could have the 1 boundary condition specified at x equal to 0.

Let us say for example, y is specified at x equal to 0 and z is specified at x equal to 2 or y prime is specified at x equal to 2 in that kind of cases, which is called the boundary value problem. We can still in some of the problems we can still use these techniques and those are called normally the shooting method in which what we do is we again convert this into an iterative scheme. So we start with so let us look at a particular example, so let us say we are given x equal to 0, y equal to 0 and x equal to 2y equal to some y_1 okay, z equal to some z_1 that is the boundary condition given it could be even y.

So but let us say 1 end we are given the derivative and other end we are given the um the y value it could be that both ends of y values are specified 1 boundary condition is specified at x equal to 0 and other boundary condition is specified at x equal to 2. So in that case what we could do is um we could use exactly the same method you convert this into 2 first order equations.

(Refer Slide time: 46:59)

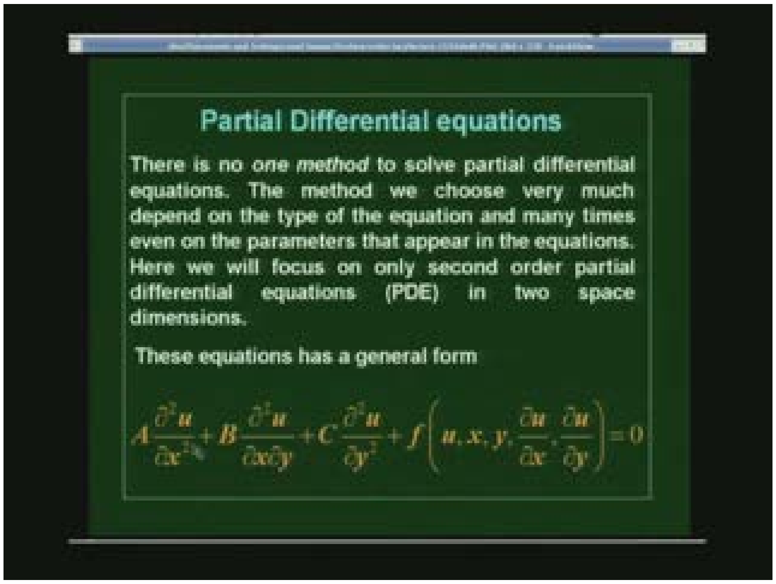


Now to use the Runge Kutta scheme as you see that we always use a scheme like this that is we are starting from y_0 and goes through some value and go to the next value this is at x plus x equal to x_0 and then go to x_0 plus h. So we are always using a scheme like that, that is we are using the y at x plus h as y at x plus 1 by 6 times k_{y1} plus k_{y2} plus k_{y3}, $2k_{y1}$ plus $2k_{y2}$ plus k_{y3}, y_2 y_3 and y_4 . Okay and 1 by 6that is the kind of scheme, we are using so to go to x plus h we need xy at x, so similarly for z so if you want to use that if you have boundary conditions specified at the 2 ends then we cannot use this because we do not know if this x value is not specified then you cannot use this.

So that cases what do you do is you make a guess you say that okay I guess I make a guess for what the y values are y_0 and y_1 are at x equal to x_0 only 1 is given other one, I make a guess and then I go all the way to the other end and then I compare the y value I obtained at the other end satisfy the boundary conditions. So that is called a shooting method so you make a guess at the initial value and then you go to the other end if you cannot do this calculation because you did not know what these initial values were and you knew only 1 initial value and the other value was actually the other boundary condition is specified at the other end and then I make a guess for this and then I go all the way in this particular case it is actually making a guess for this because I have been given z at the x equal to 2 not at x equal to 0.

So I make a guess for this value at x equal to 0 and then I go all the way to x equal to 2 and then I see the value y obtained on that side is the same as this or not and if it is not I make an adjustment into our initial guesses again and then I go there again to that point.
So in this iterative scheme I can actually obtain a solution to that equation a little more tedious and but sometimes not too inefficient method called shooting method.

(Refer Slide time: 49:27)



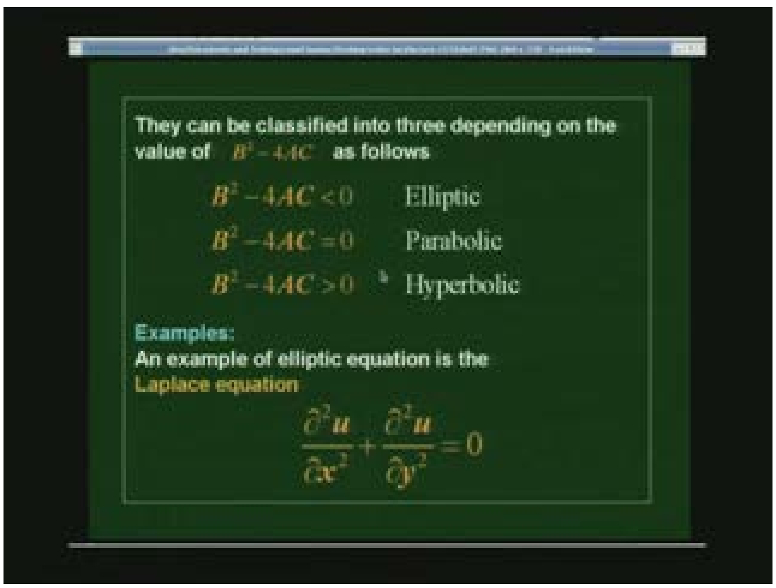
So with this we stop the discussion on the ordinary differential equations and now we would go over to a discussion on partial differential equations that will be our next topic to look at. So far we have only looked at equations which are of 1 variable type a derivative with respect to only one independent variable, we have only everything as a function of x alone. So both z that is y prime and y in this particular case were the function of x alone but we could have them as functions of many variables, many independent variables.

In that case this is differential equations would have derivative with respect to all those independent variables for example, here is a general partial differential equation it is a function variable u which is what we want to find and u and its derivatives as what we want to find, so if you want to solve this equation what we mean by that is we want to find out u as a function of x and y. So x and y are both independent variables, so you want to find u as a function of x and y and derivative of u as a function of x with respect to as function of x and y and derivative of u as a function of y with respect to x and y. So that will be the, that would be what you mean by finding the solution of this partial differential equation.

So how do we go about doing that. That will be the discussion which will be which we will do now just add the general partial differential equation in this form with coefficients a, b and c they are some constant coefficients a, b and c and we have a differential equation of this form and then some function of and this here some function of u x y derivative of u derivative of y etcetera.

So we have a general differential equation of this form. So remember that this can be any function of u x y derivative of u with respect to x derivative of u with respect to y and then we can classify this differential equation into 3 categories called elliptical parabolic and hyperbolic, not that we need this classification to solve this equation but we will try to do 1 example or look at 1 example from each of these classes and then how to solve that and there is no, one thing we should keep in mind is that there is no general method for each class of this equation.

(Refer Slide time: 51:14)



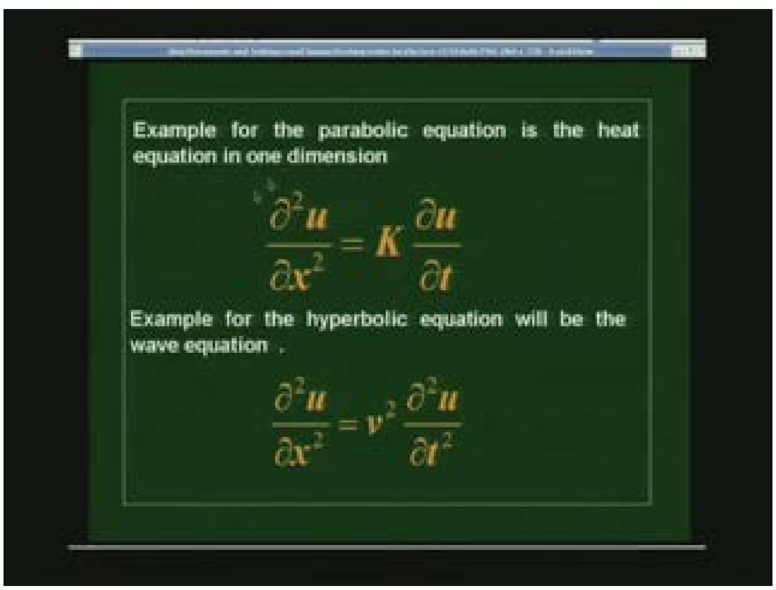
It is not that if we could classify a differential equation into an elliptic form then there is partial differential equation into as an elliptic partial differential equation and there is 1 numerical method and then there is a parabolic equation there is an another numerical method etcetera. They do not exist actually for the difficulty with partial differential equations is that the method which we use to solve partial differential equation is specific to a particular equation, there is there are a very few methods which are general enough so that you can use it for a class of problems.

So the technique which we use to solve partial differential equations depends very much on the sometimes even the coefficient sitting in front of these derivatives the range of those coefficients etcetera. We will look at that in the coming classes here, let us just see this as 3 different classes so that b square minus 4 a c is less than 0 as elliptic for example, the equation would be laplace’s equation which you are familiar with in electrostatics etcetera.

So this laplace’s equation is a case of elliptic partial differential equation remember a b and c, so a, b and c are the coefficients of the 2 second derivatives and a and c are the coefficients of the 2 second derivatives with respect to variable x and y respectively and the b is the coefficient of, cross second derivative del square u by del x del y. So in the case of laplace’s equation we have a as 1 and c as 1, b is 0 and f this function is 0 that is what laplace’s equation is and then we have that as satisfying this as b is 0.

So we have a c is less than 0a and c are the same sign, so a c is less than 0 so it is an elliptic equation if you had a minus sign here that would be a case where again b is 0 but a and c have now minus opposite signs. So it will be a positive case so that will be case of hyperbolic equation.

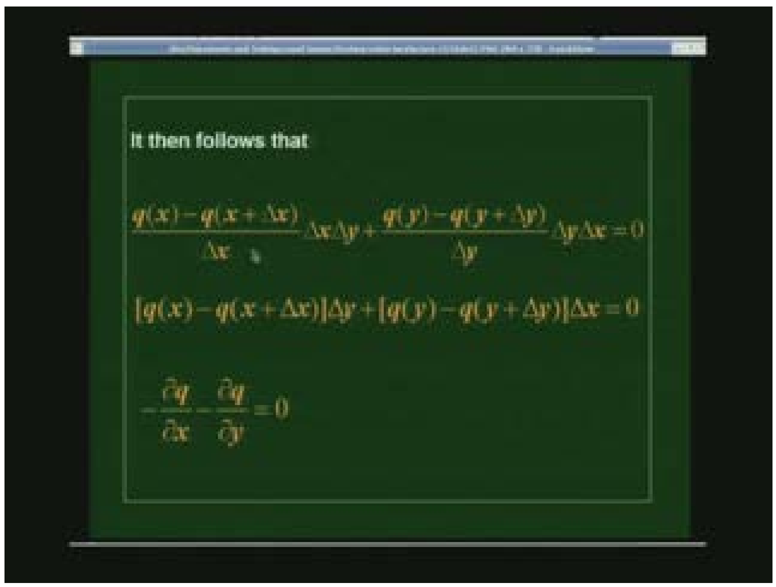
(Refer Slide time: 54:10)



So we will see the wave equation as an example of, so now if you had one of them 0, that is we have only 1 derivative with respect to we only had this derivative and we did not have the second derivative with respect to y but we had some first derivative with respect to y that is in this equation if you have only 1 of them a or c as non 0 and b is also 0 and then some function here which is non 0 function that kind of equations would be parabolic equations. Okay as I said equation for an elliptic equation is an example for an elliptic equation is laplace’s equation and an example for a parabolic equation here is a parabolic equation is a heat equation that is where b square minus 4 a c is 0, that is the heat equation or the diffusion equation.

We will solve, we will look at solutions of this equation also and then an hyperbolic equation would be the wave equation. So that is the 3 different examples which you would be interested in looking at so remember laplace's equation as an example of elliptic equation and heat equation or a diffusion equation as an example of parabolic equation and a wave equation as an example of a hyperbolic equation. So one other thing which we always do in the methods which we use here that is heat is not a general method for solving partial differential equation unlike Runge Kutta scheme, which can be used for a class of ordinary differential equation there is no 1 method to solve a partial differential equation but the method which we would be looking at here is mostly using, replacing the derivatives by its discrete form.

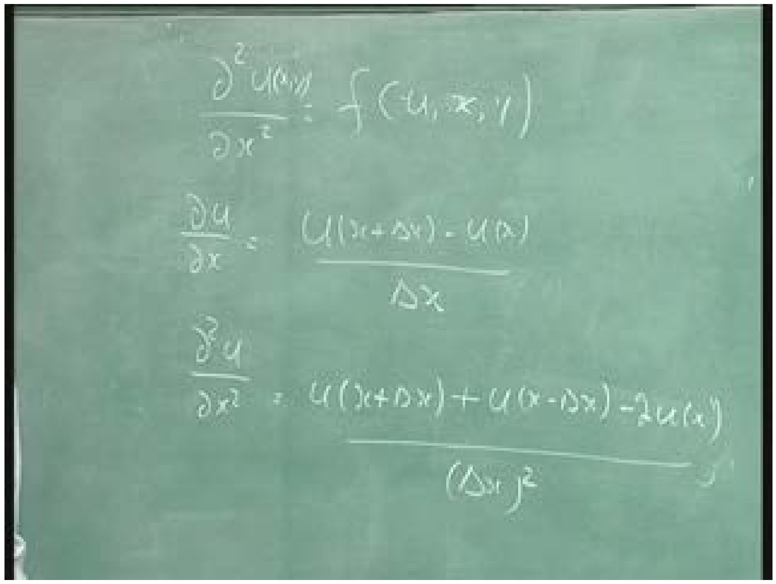
(Refer Slide time: 55:12)



So that is we write an equation of this form and then we would replace the derivatives by its difference equation. So that is something which we would be looking at hence, okay so that is we would have let us say we have an equation of del square u by del x square with some function of x some u and x and y you know. Okay so some equation like this so u x and y x is a function of x and y. So and then we would replace we would replace this derivative by its discrete derivative that is we would write del u by del x for example, you could write it as u at x plus delta x minus u at x divided by delta x.

So that will be a simple forward differencing scheme and similarly we could write del square u by del x square as u at x plus delta x plus u at x minus delta x minus 2u of x divided by delta x square and similarly for y, and these we will replace these difference equations into this equation into this differential equation here and we will convert this differential equation to a difference equation and solve for these u values at various x values and y values. Okay that is what it is something that we will be doing as a method for solving partial differential equations.

(Refer Slide time: 56:54)



We will not be looking at more complicated methods for solving partial differential equations in this course. So we will such techniques to solve both laplace's equation and heat equation in the coming few lectures.