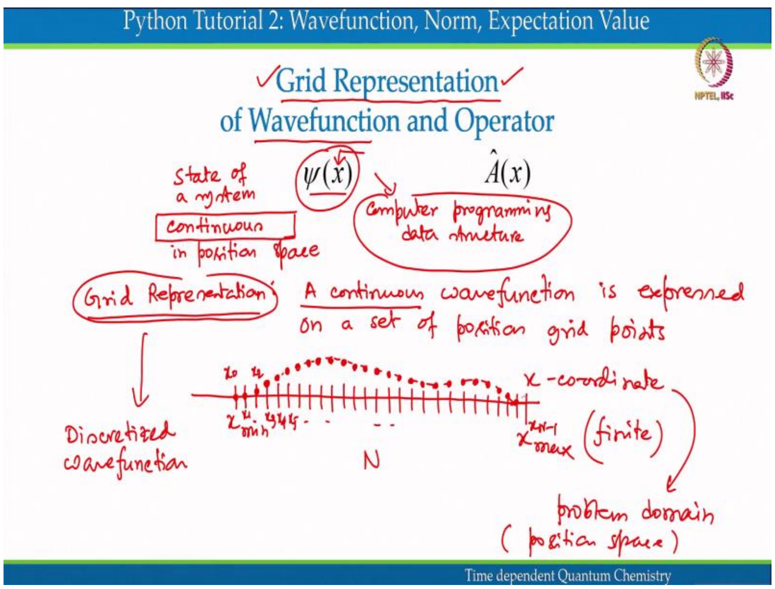
Time Dependent Quantum Chemistry
Professor. Atanu Bhattacharya
Department of Inorganic and Physical Chemistry
Indian Institute of Science, Bengaluru
Lecture 14
Grid Representation of Wavefunction

Welcome to python tutorial 2 of the course Time Dependent Quantum Chemistry. In the earlier python tutorial, we have learned different small programming with python, we have used arithmetic computation and we have shown a few loops and how to plot a graph.

Now in this tutorial, we will go for how to represent a wave function and later we will find out how to find represent an operator but here we will show how to represent a wave function in python programming and how to get the norm which we need to normalize the wave function and also we will calculate the expectation value. So, let us go over this tutorial

(Refer Slide Time: 1:18)



The wave function and the operator with function and the operator, these are the two key constituents of quantum mechanics. The wave function represents the state of the system which is represented by $\psi(x)$, the state of our system is represented by the wave function. On the other hand, when operator acts on the wave function we get the observable so when this operator acting on the wave function we get the observable which is which can be observed experimentally.

Therefore, because these are the two constituents of quantum mechanics, the first step towards obtaining numerical solution of any quantum mechanical problem is to represent the wave function and the operator in appropriate computer programming data structure, that is the first step we should have, if we want to use numerical solution of quantum mechanical problem, we have to convert this wave function and operator in the computer programming data structure.

So this question is how to represent them, almost all currently available numerical methods for solving quantum mechanical problems make use of grid representation of the wave function and the operator. So, one way to do this is the grid representation, what is grid representation? I will present it a wave function by its nature is continuous.

So, what would be the procedure to transform this operator into computer programming structure this is something which will look at later, not now in this not in this tutorial, we just stay focused on the wave function part. So, operator we will be dealing with operator later. But, we know that wave function by its nature is continuous, that is the way we have made the postulate of quantum mechanics it is a continuous function continuous in the coordinate space.

So, whatever variable we use here we are using x as a variable which is the position space, so it is going to be continuous in the position space. And this property of an wave function comes from the postulate of quantum mechanics. So, it is continuous in position space. But in the grid representation, the representation which will use to represent a wave function in the computer programming data structure that is the grid representation in the grid representation what will do a continuous wave function is a continuous wave function, so grid representation is following a continuous wave function is expressed on a set of position grid points.

So, what does it mean? The entire position space this is the x axis entire position space theoretically x axis goes from minus infinity to plus infinity but in computer programming data structure will not be able to use this infinite limit, so what will do we will consider one x minimum and one x maximum value, within this range we will express the wave function and we will assume that in this boundary x minimum and x maximum wave function becomes 0.

So, because it is 0 here already so it will be 0 at minus infinity and plus infinity as well. So, we are following the boundary condition in the computer programming. So, the boundary has to be selected such a way the finite boundary this is the finite boundary you are selecting in the x axis this is the x coordinate, so finite boundary has to be selected such a way that the wave function will become 0 before that boundary.

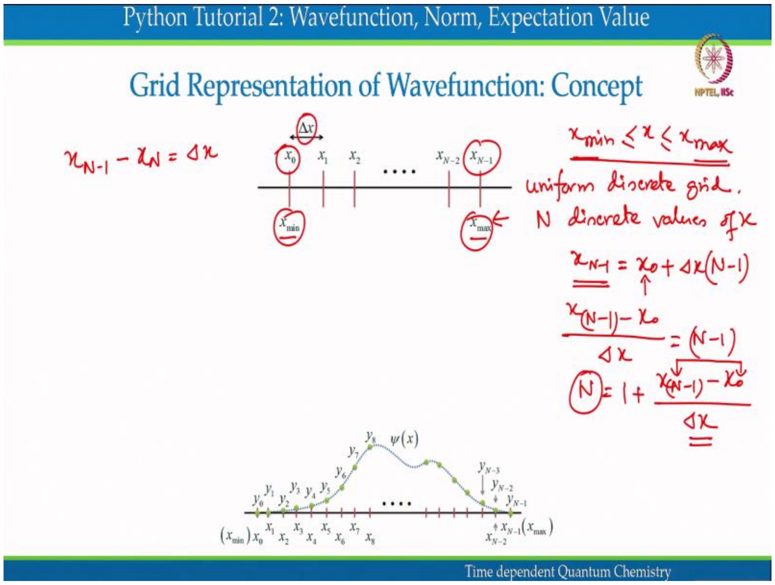
So, within this boundary what will do, we will not be able to represent it a continuous wave function we have to divide the entire problem domain, this is called problem domain the this this x coordinate it is also position space is also called problem domain or the position space this entire position space will be divided we will be dividing it into an uniform grid.

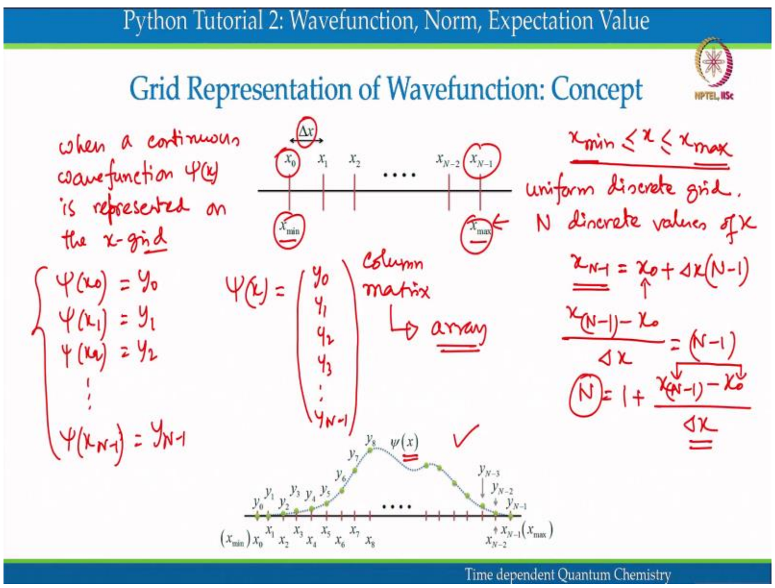
So, we will have a discrete value of x and corresponding to discrete value of x will have discrete value of the wave function and each one let us say this is x0, this is x1, this is x2, this is x3, x4, x5 like this way and then this is if I have n number of such grid points then I have this is xN-1 because it is starting from x0 that is why it is going up to xN-1, that is the maximum value of x we get.

So what we do here, this in under grid representation we actually discretize the continuous wave function. A wave function by its nature but from the postulate of quantum mechanics is has to be continuous but we have to represent in the computer programming we have to represent a continuous wave function in a discretized wave function. So, grid representation gives me a discretized wave function, so discrete values of x I will get.

So, instead of drawing this I have to draw it like this way it was like this and I have now discrete values of wave function and this is going to represent the wave function in the python programming I get discrete values on the x grid.

(Refer Slide Time: 10:06)





So, to understand this discretization of wave function further, we have to divide the certain range of x coordinate as I said that this x coordinate I cannot take minus infinity to plus infinity limit in computer programming, I have to take the finite limit x minimum and x maximum and this range so x has to be then x minimum less than equals x maximum. So, x value would be in between this x minimum and x maximum value and will make sure that when you are defining this finite domain we have to make sure that the wave function becomes 0 at the boundaries as if it already has become 0.

So, the idea is that at minus infinity and plus infinity it will be 0 and the requirement for wave function to be 0 at 0 value at minus infinity and plus infinity comes from the fact that this wave function should live in the Hilbert space we have studied this Hilbert space meaning of Hilbert space in module 4 where we have discussed that every wave function has to be square integrable, so that we can normalize it.

So, the entire x axis will be will be divided by a suitable small step size l deltax

to produce the uniform discrete grid, so this is the uniform discrete grid we produce. So, we have let us say n number of discrete values of discrete values of x within this range

$$x_{\min} \leq x \leq x_{\max}$$

we have n number of discrete values so one can write down here is that xN-1 - xN equals delta x that is the separation between each grid points.

$$x_{N-1} - x_N = \Delta x$$

$$x_{N-1} = x_0 + \Delta x(N-1)$$

So, if we rearrange it I can write down

$$\frac{x_{N-1} - x_0}{\Delta x} = (N-1)$$

$$N = 1 + \frac{x_{N-1} - x_0}{\Delta x}$$

So, how many grid points I have it depends on the spacing I am selecting and the range I am selecting this is the range we have selected xN-1 that is the maximum range and this is the minimum value in the range.

So, this many number of grid points we have and because we have this many grid points will have that many values of the function and this is exactly what we have shown here a continuous wave function which is represented by dot line here but we have to represent it on this grid. So, on each grid point I get the value of the function.

So, when a continuous wave function is represented on the x grid, we get this values

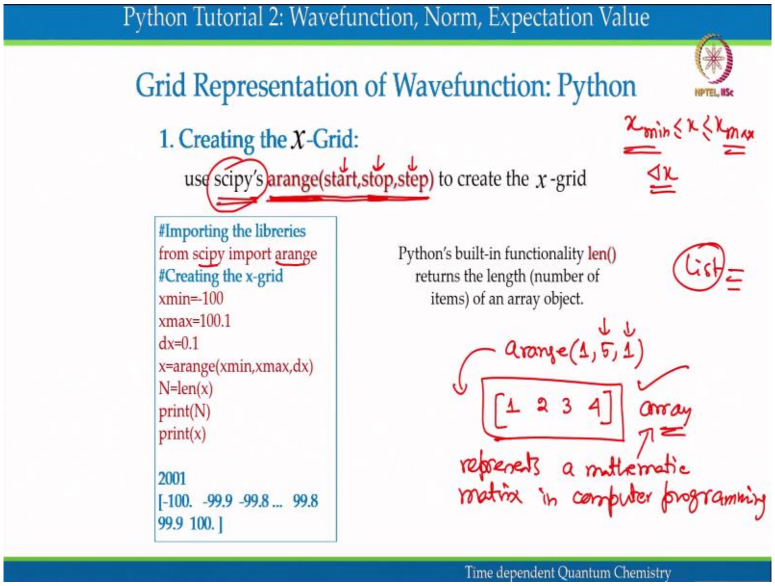
$$\psi(x_0) = y_0$$
$$\psi(x_1) = y_1$$
$$\psi(x_2) = y_2$$
$$.$$
$$.$$
$$\psi(x_{N-1}) = y_{N-1}$$

So, these are the values we get and in linear algebra one can represent this entire set of values as a column matrix.

$$\psi(x) = \begin{pmatrix} y_0 \\ y_1 \\ . \\ . \\ y_{N-2} \\ y_{N-1} \end{pmatrix}$$

We form and that is the representation and this matrix this is a column matrix, this column matrix can be represented in the python programming by an array. So, pretty much we have got an idea how we are going to represent the entire wave function in the computer programming we have to prepare the array.

(Refer Slide Time: 17:22)



So, before we prepare the array, first we have to create the x grid that is the way we have said that we have to prepare the x grid where we have to define the certain range which is let us say minimum to maximum, this is the range I have to select and then after selecting range I have to select also delta x, then I can create this x grid. And on the x grid we have to find out the wave function values to represent the wave function.

We have already realized from python tutorial the first python tutorial that this arange functionality we have already used arange functionality start stop step of psi pi this scipy module we are using it can actually return a list of evenly spaced discrete values of x within the given range. So, this is the range start stop that is the range and then and then this is the difference between each grid point each points evenly space points we can get it.

So, we will use this arange functionality to create that and I just remind that in arange functionality when will use that stop this part will not be used in the sequence or will not be included in the sequence. So, arange if I write arrange(1,5,1), it will return me evenly spaced discrete values of the discrete values of x in a following way.

It will give me 1 then it will give me 2 because the difference I have selected to be 1 then it will give me 3 then it will be 4 then it is supposed to give me 5 but because 5 is stop and the way this arrange functionality works it will just exclude the last one. So, I get back this as a list of values, [1,2,3,4].

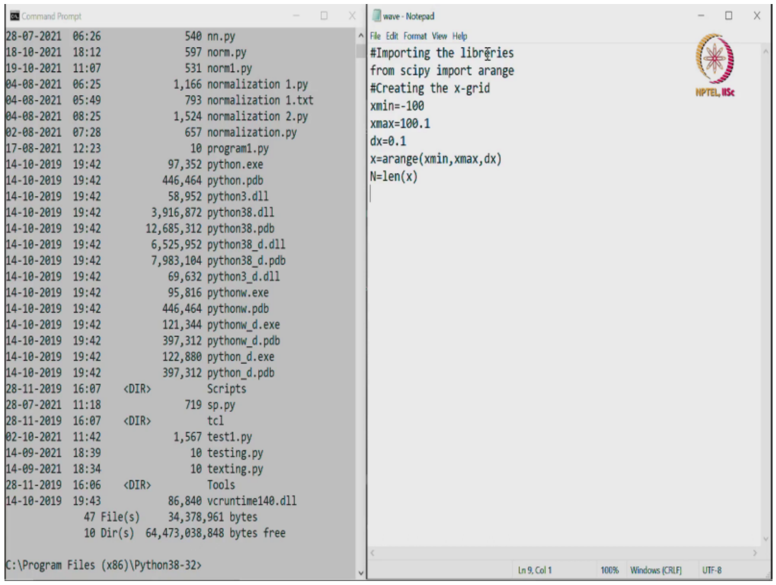
To be technically correct although we should remember that we have also briefly mentioned that in the previous python tutorial that arange functionality returns an array not a list so although we are calling it as a list but technically this is an array we get. So, and the difference between list and an array in python programming the list program this list is actually python's built-in program which is present in python programming but arange functionality this comes from scipy this is actually scipy's functionality.

Now this list functionality of python's built-in functionality list cannot be used for linear algebra routines which are implemented in scipy, for that we have to use always this array, so that is the reason why we are using arange functionality it returns an array. And one dimensional array is nothing but the column matrix or it can also be represented as a row matrix as well.

So, we will use this arange functionality in general the data structure which is used to represent mathematical matrix in computer programming is called array. So, this is the definition of array is it represents a mathematical matrix in computer programming. And array can be N dimensional it can have dimensionality N but two-dimensional array is called matrix and a one-dimensional array is called a vector this is an one-dimensional array which we have prepared for the x grid.

Python does not have intrinsic functionality to deal with the array or different array operations broadly linear algebra routines for that one needs to import ah import scientific functionality specific functionality from scipy. So, that is the reason why you are using scipy so you have to import this arange functionality from scipy. So we will prepare the grid and grid preparation is very simple we will move to laptop now and will prepare the grid.

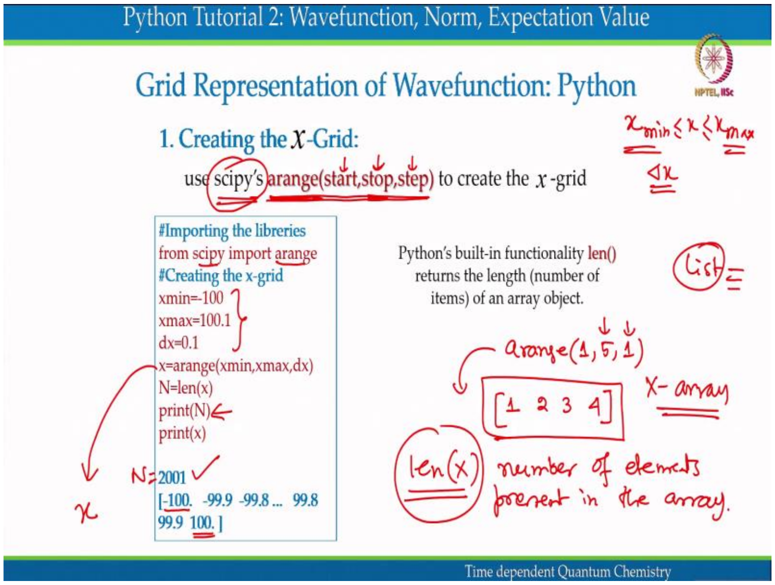
(Refer Slide Time: 22:50)

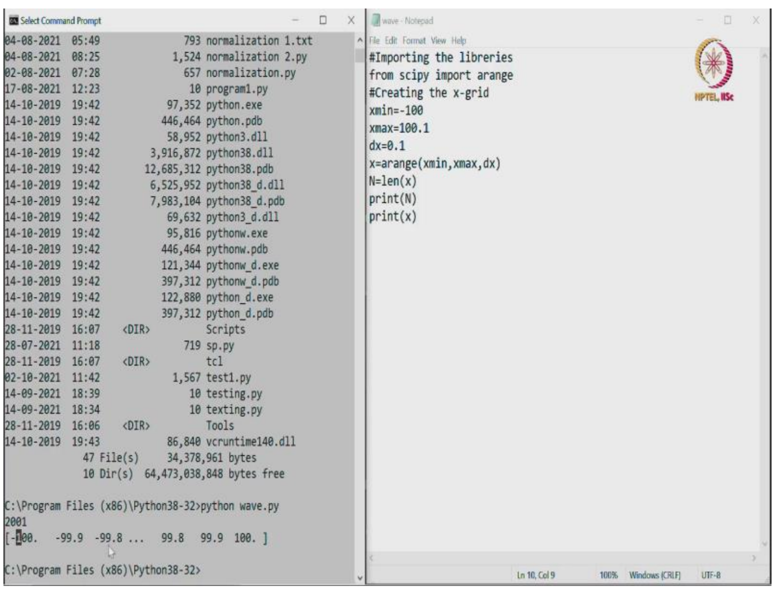


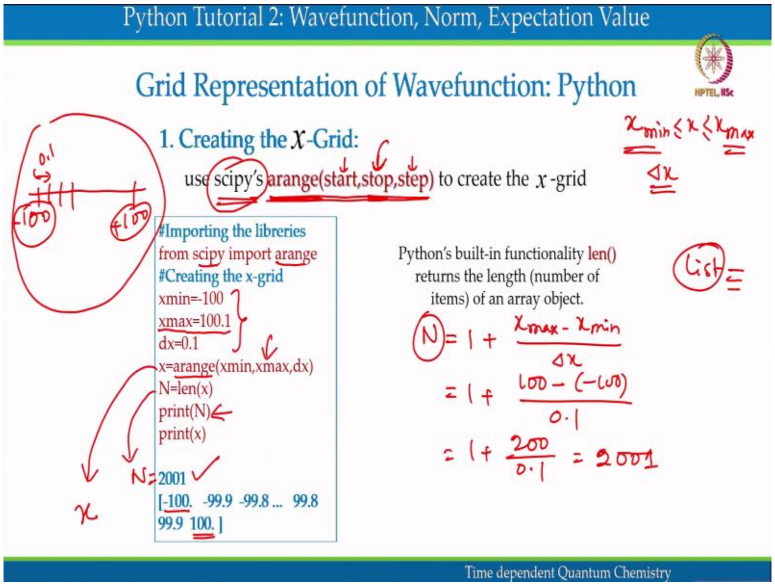
We will first import and this hash will indicate that anything written after the hash would not be executed in the python programming it is just for our information. So, we are importing the required libraries we will name it wave dot pi and from scipy import arrange functionality. Then, we will create the x grid for that we have to define x minimum x minimum we are defining 100 x maximum remember we are defining 100.1 because this point will be excluded from the list and dx that is the difference between grid point adjacent grid point is 0.1.

That is why we have selected 100.1 so that the entire range is going to be minus 100 to plus 100 this is this minimum is minus 100 and maximum is plus 100. Then, we will just specify x equals arange x minimum x maximum dx that is the construct of the arrange functionality then we will just write down n equals len(x), what it does it will try to find out the number of elements I will move to the presentation in the slide.

(Refer Slide Time: 25:11)







So, this length is a python's built in functionality which is len(x) so what it does if x is an array then it will find out number of elements present in the array that is the functionality of length this is python's built-in functionality one does not need to import it from anywhere will move to the laptop and then I will just print n, I will print n to check how many elements I have in the prepared grid and I will also print x that is the x grid I have.

So, if I want to now run the program, I have to use python then we have given wave name dot pi dot by extension has been given and we can run the program, when you run the program we see that we are getting 2001 number of elements this is the first print n that is the first printing and second printing is that the values of the x grid values of the x grid is minus 100 then minus 99.9 then minus 99.8 that is the way values should be.

So, we have we will move to this slide we have 2001 number of n so n equals 2001 and I am printing the entire x array here and entire x means grid points I am writing and if I look at the grid points it is starting from minus 100 to and ending at plus 100 with us spacing between adjacent grid points is 0.1 that is exactly what we wanted to do this is the construct which we have used already.

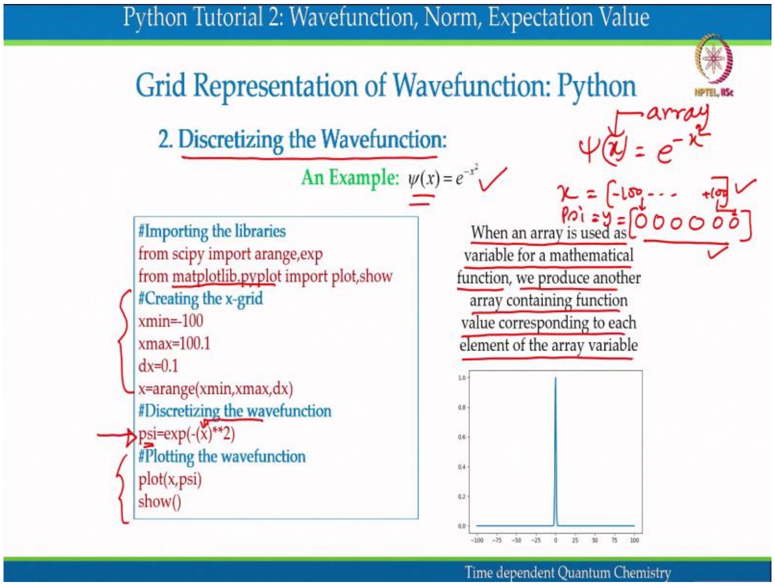
Now we can recheck whenever we are doing some numerical method we are following if we are following numerical methods in the beginning we should always try to match the result with an analytical solution. So, we have already seen that the n can be calculated n is nothing but number of elements I will have in the grid point, so I have prepared the grid points from minus 100 to plus 100 with the separation 0.1 that is the grid points x grid I have prepared.

So, how many elements I have that is given by

$$
\begin{aligned}
N &= 1 + \frac{x_{\max} - x_{\min}}{\Delta x} \\
&= 1 + \frac{100 - (-100)}{0.1} \\
&= 1 + \frac{200}{0.1} \\
&= 2001
\end{aligned}
$$

And to make this range to be minus 100 to plus 100 I just remind that x maximum we have taken to be 100.1 because the last in the that is the way arrange functionality works a range functionality this stop part this stop will not be included in the sequence that is the way it the constant it works. So, once we have already discretized the wave once we have a x grid on this x grid will now represent or discretize the wave function.

(Refer Slide Time: 29:51)



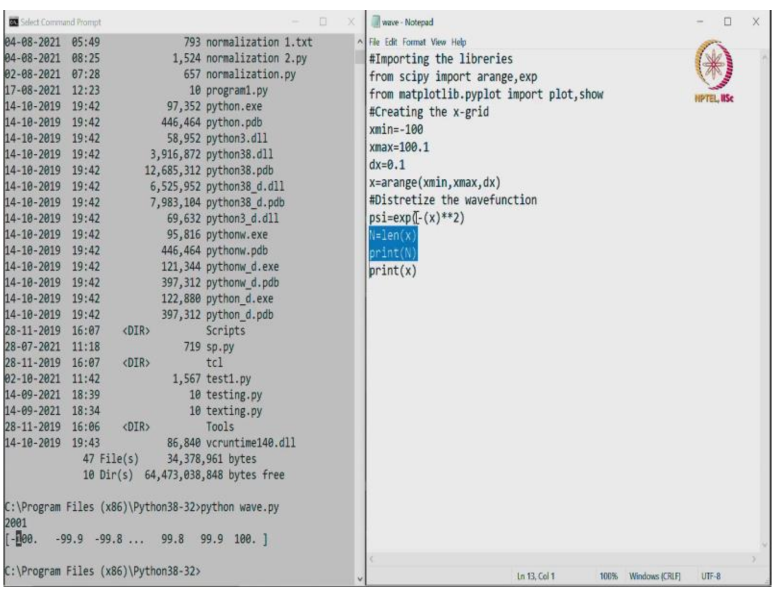
So, when you are discretizing a wave function, you have to take one example one can take an example of let us say Gaussian function it is an it is just an wave function that is the one representation of the wave function a Gaussian function and we will take this example to demonstrate that how to discretize the wave function on the x grid.

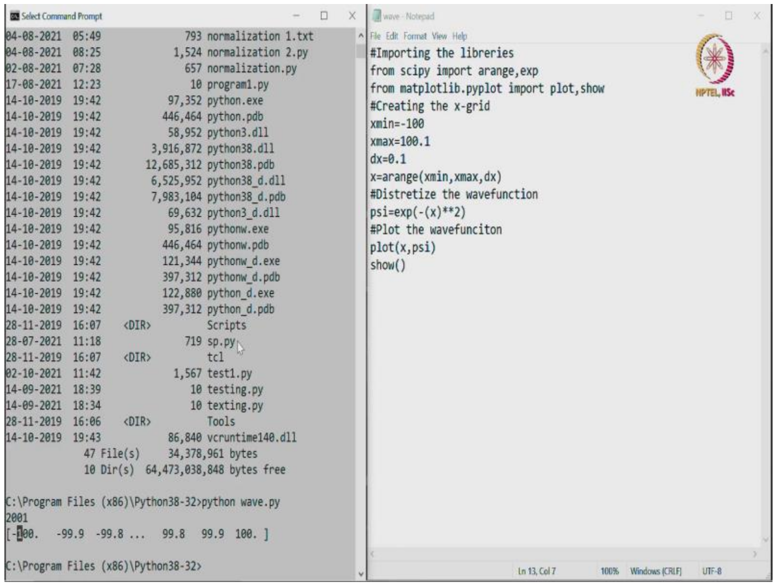
$$\psi(x) = e^{-x^2}$$

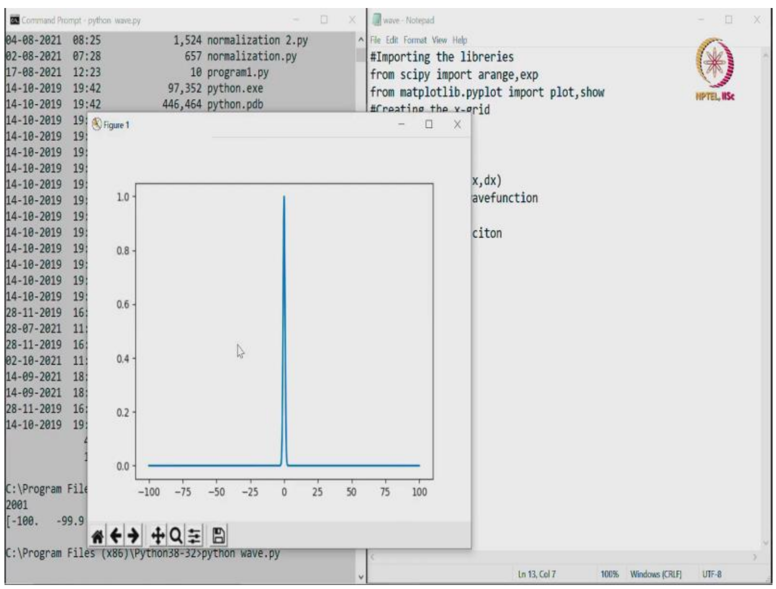
So, the first part of this programming is quite understandable now we have created the x grid first and then we have to discretize it. When you are discretizing it, it is very simple when an array is used as variable for a mathematical function mathematical function such as this psi x. So, $\psi(x)$, x is an array if this is an array then we produce if we directly write down $\psi(x)$ equals e^{-x^2} then we produce another array containing function value corresponding to each element of the array variable. So, basically if we so this construct directly representing this construct because x is an array I will produce another array of $\psi(x)$.

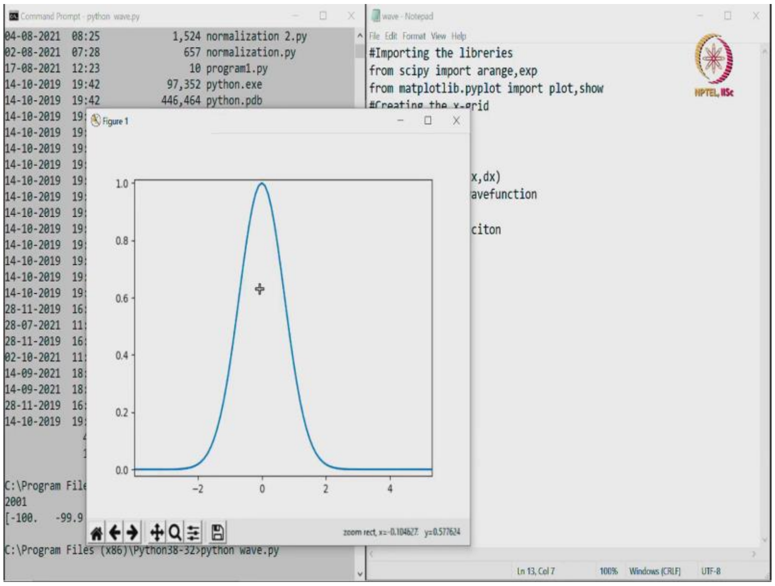
So, if x is an array which is starting from 100 to plus 100, then I will get psi which can also be expressed as y is also an array and this array will have the elements where function values will be presented corresponding to each grid point that is the way it works quickly and this plotting part we are familiar with this plotting part with using this matplot library pi plot sub module we are plotting it to make sure that we have the wave function the desired wave function.

(Refer Slide Time: 32:08)




Select Command Prompt
04-08-2021 05:49 793 normalization 1.txt
04-08-2021 08:25 1,524 normalization 2.py
02-08-2021 07:28 657 normalization.py
17-08-2021 12:23 10 program1.py
14-10-2019 19:42 97,352 python.exe
14-10-2019 19:42 446,464 python.pdb
14-10-2019 19:42 58,952 python3.dll
14-10-2019 19:42 3,916,872 python38.dll
14-10-2019 19:42 12,685,312 python38.pdb
14-10-2019 19:42 6,525,952 python38_d.dll
14-10-2019 19:42 7,983,104 python38_d.pdb
14-10-2019 19:42 69,632 python3_d.dll
14-10-2019 19:42 95,816 pythonw.exe
14-10-2019 19:42 446,464 pythonw.pdb
14-10-2019 19:42 121,344 pythonw_d.exe
14-10-2019 19:42 397,312 pythonw_d.pdb
14-10-2019 19:42 122,880 python_d.exe
14-10-2019 19:42 397,312 python_d.pdb
28-11-2019 16:07 <DIR> Scripts
28-07-2021 11:18 719 sp.py
28-11-2019 16:07 <DIR> tcl
02-10-2021 11:42 1,567 test1.py
14-09-2021 18:39 10 testing.py
14-09-2021 18:34 10 texting.py
28-11-2019 16:06 <DIR> Tools
14-10-2019 19:43 86,840 vcruntime140.dll
47 File(s) 34,378,961 bytes
10 Dir(s) 64,473,038,848 bytes free
C:\Program Files (x86)\Python38-32>python wave.py
2001
[-100. -99.9 -99.8 ... 99.8 99.9 100.]
C:\Program Files (x86)\Python38-32>
wave - Notepad
File Edit Format View Help
#Importing the libreries
from scipy import arange,exp
from matplotlib.pyplot import plot,show
#Creating the x-grid
xmin=-100
xmax=100.1
dx=0.1
x=arange(xmin,xmax,dx)
#Distretize the wavefunction
psi=exp(-(x)**2)
#Plot the wavefunciton
plot(x,psi)
show()
NPTEL, IISc
Ln 13, Col 7
100%
Windows (CRLF)
UTF-8


Command Prompt - python wave.py
04-08-2021 08:25 1,524 normalization 2.py
02-08-2021 07:28 657 normalization.py
17-08-2021 12:23 10 program1.py
14-10-2019 19:42 97,352 python.exe
14-10-2019 19:42 446,464 python.pdb
Figure 1
1.0
0.8
0.6
0.4
0.2
0.0
−100 −75 −50 −25 0 25 50 75 100
2001
[-100. -99.9
C:\Program Files (x86)\Python38-32>python wave.py
wave - Notepad
File Edit Format View Help
#Importing the libreries
from scipy import arange,exp
from matplotlib.pyplot import plot,show
x,dx)
avefunction
citon
NPTEL, IISc
Ln 13, Col 7
100%
Windows (CRLF)
UTF-8



So, we will move to laptop and we will now delete will not delete this part will just keep it as it is importing the libraries now, we will add one more line here because we are going to plot it that is why we are importing plot and show functionality from matplot library pi plot so from matplot library dot matplotlib dot pi plot import plot and show both I need.

And then I have created this grid this is this part is creating the grid and then I will now discretize I will now discretize the wave function and this is very simply done I can name it psi psi I could name it y or something else as well just I am naming it psi psi equals this is exponential. So, exponential is not available this mathematical function is not available with python it has to be imported from scipy so I have to import from scipy. So, I have written so psi so I am importing this exponential from scipy and then now this psi will be the psi will be defined psi is defined as minus x square is double star 2.

This is the psi and next what we would like to do is that we would like to plot, so that I can see what I have discretized so plot the wave function I can plot it plot construct is x comma it cannot be y the name is psi so x , psi and that is the construct is it will plot and it will take the x array and y array and corresponding elementwise they will plot it and finally I have to give this show command to display the wave function.

So, if I now run this program I get the wave function one can select a particular x limit to show the Gaussian function for us shorter x limit because it will give you the correct shape of the

Gaussian function otherwise if we see it in a longer x scale it is it may not give you the right representation it is just a representation problem but it plots the correct wave function.

So, we have this discretized wave function plotted already and this psi we will move to the slide right now so this psi has now an array so both x and psi is an array so the idea of discretizing the wave function is to prepare the array associated with the wave function on the grid points. So, once we have represented a wave function here we have taken an example of Gaussian wave function we can go ahead and normalize the discretized wave function.

We will continue the session will continue learning how to represent the wave function how to normalize wave function how to normalize a discretized wave function ah in the next session.