A network graph with white nodes and edges on a teal background. The graph consists of numerous interconnected nodes, forming a complex, multi-layered structure. The nodes are small white circles, and the edges are thin white lines. The overall shape of the graph is somewhat irregular, with a dense central core and more sparse, branching structures extending outwards. The background is a solid teal color with some faint, glowing blue spots and a subtle bokeh effect.

Python Programming Part 3

Instructor: Vision Wang

Email: xinwang35314@gmail.com

Part 3: Python Standard Library

- Mathematics
- Statistics
- Dates and Times
- Modules for AI



Python – Math Module

Some of the most popular mathematical functions are defined in the math module. These include trigonometric functions, representation functions, logarithmic functions, angle conversion functions, etc.

Pie (π) is a well-known mathematical constant, which is defined as the ratio of the circumference to the diameter of a circle.

```
>>> import math
>>> math.pi
3.141592653589793
```

Euler's number is a base of the natural logarithm.

```
>>> math.e
2.718281828459045
```

Question: calculate using
Python: e^{10}

Python – Math Module

- The math module contains functions for calculating various trigonometric ratios for a given angle.
- The functions (sin, cos, tan, etc.) need the angle in radians as an argument.
- **degrees()** : to convert the angle from radians to degrees. **radians()**: to convert the angle from degrees to radians.

```
>>>math.radians(30)
0.5235987755982988
>>>math.degrees(math.pi/6)
29.999999999999996
```

- The following statements show **sin, cos and tan** ratios for the angle of 30 degrees (0.5235987755982988 radians).

```
>>math.sin(0.5235987755982988)
0.49999999999999994
>>>math.cos(0.5235987755982988)
0.8660254037844387
>>>math.tan(0.5235987755982988)
0.5773502691896257
```

Python – Math Module

- **math.log()** method returns the natural logarithm of a given number. The natural logarithm is calculated to the base e.

```
>>>math.log(20)
2.995732273553991
```

- **math.log10()** method returns the base-10 logarithm of the given number. It is called the standard logarithm.

```
>>> math.log10(100)
2.0
```

- **math.exp()** method returns a float number after raising e (math.e) to given number. In other words, $\text{exp}(x)$ gives $e^{**}x$.

```
>>> math.exp(13)
442413.3920089205
```

Python – Math Module

- **math.pow()** method receives two float arguments, raises the first to the second and returns the result. In other words, `pow(4,4)` is equivalent to `4**4`.

```
>>> math.pow(3,4)
81.0
>>> 3**4
81
```

- **math.sqrt()** method returns the square root of a given number.

```
>>> math.sqrt(100)
10.0
>>> math.sqrt(2)
1.4142135623730951
```

Python – Math Module

- **ceil()** function approximates the given number to the smallest integer, greater than or equal to the given floating point number.
- **floor()** function returns the largest integer less than or equal to the given number.

```
>>> math.ceil(6.5867)
```

```
7
```

```
>>> math.floor(6.1345)
```

```
6
```

Python – Statistics Module

- The statistics module provides functions to mathematical statistics of numeric data.
- **mean()** method calculates the arithmetic mean of the numbers in a list.

```
>>> import statistics
>>> statistics.mean([134,45,78,90])
86.75
```

- **median()** method returns the middle value of numeric data in a list.

```
>>> statistics.median([134,45,78,90])
84.0
>>> statistics.median([134,45,65,78,90])
78
```

- **mode()** method returns the most common data point in the list.

```
>>> statistics.mode([134,45,65,78,90,45,32,45])
45
```


Python – Dates and Times

- **datetime** module supplies classes for manipulating dates and times in both simple and complex ways.

```
>>> from datetime import date
>>> now = date.today()
>>> print(now)
```

```
>>> birthday = date(1964,7,31)
>>> age = now - birthday
>>> age.days
20373
```

Modules for AI

AI Application	Python Module
Data analysis and visualization	NumPy, SciPy, Pandas, Seaborn
Machine learning	TensorFlow, Keras, Scikit-learn
Computer vision	OpenCV
Natural language processing	NLTK, spaCy

Modules for AI

Python | Text to Speech conversion

- **pyttsx3** is a text-to-speech conversion library in Python.
- To install the pyttsx3 module, first of all, you have to open the terminal and write

```
> pip install pyttsx3
```

Try this module using the codes below and see what you will get.

```
# Import the required module for text-to-speech conversion
import pyttsx3

# init function to get an engine instance for the speech synthesis
engine = pyttsx3.init()

# say method on the engine that passing input text to be spoken
engine.say('Hello, Jason. How can I help you?')

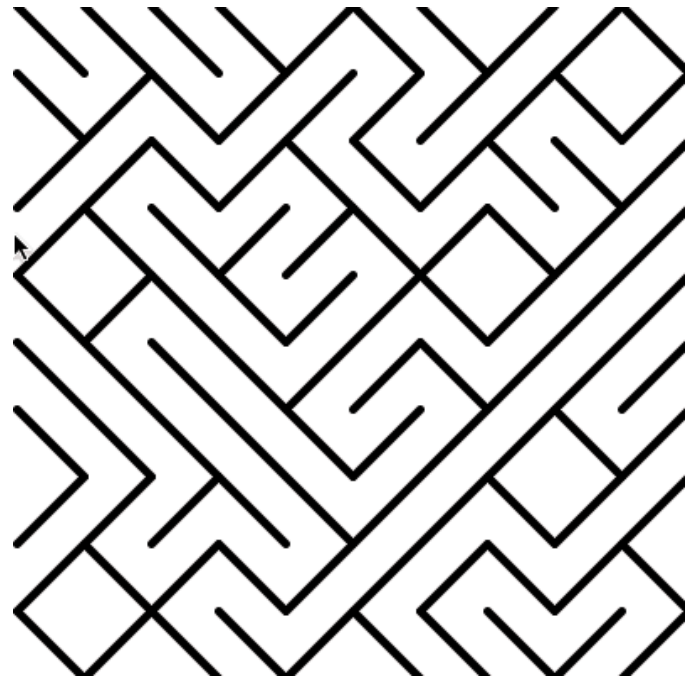
# run and wait method, it processes the voice commands.
engine.runAndWait()
```

Modules for AI

Python Games

- **Free Python Games** is an Apache2 licensed collection of free Python games intended for education and fun. The games are written in simple Python code and designed for experimentation and changes.
- Free Python Games are implemented using Python and its Turtle module.
- Installing Free Python Games is simple with pip:
 - > **pip install freegames**

Q: Try to use freegames module to realize the maze game. Revise the original code and redesign this game with more fun.



Code for maze game

```
from turtle import *
from random import random
from freegames import line

def draw():
    "Draw maze."
    color('black')
    width(5)

    for x in range(-200, 200, 40):
        for y in range(-200, 200, 40):
            if random() > 0.5:
                line(x, y, x + 40, y + 40)
            else:
                line(x, y + 40, x + 40, y)

    update()

def tap(x, y):
    "Draw line and dot for screen tap."
    if abs(x) > 198 or abs(y) > 198:
        up()
    else:
        down()

    width(2)
    color('red')
    goto(x, y)
    dot(4)

setup(420, 420, 370, 0)
hideturtle()
tracer(False)
draw()
onscreenclick(tap)
done()
```