Python #2

String Theory

david@codecachet.org
CodeCachet.org

$$G_{\mu\nu} \equiv R_{\mu\nu} - \frac{1}{2} R g_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu}$$

$$S_B = \frac{k_B 4\pi G}{\hbar c} M^2$$

$$\psi_1(x) = \frac{1}{\sqrt{k_1}} (A_- e^{ik_1 x} + A_- e^{-ik_1 x}) \quad x < 0$$

$$k_1 = \sqrt{2mE/\hbar^2}$$

$$\sigma = \frac{24\pi^3 L^2}{T^2 c^2 (1 - e^2)}$$

$$R_{\mu\nu} - \frac{1}{2} R g_{\mu\nu} + \Lambda g_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu}$$

$$H = \frac{P P}{2m} + V(r)$$

$$P = -i\hbar\nabla$$

$$S = \frac{1}{2k} \int R \sqrt{-g} d^4x$$

$$S = \frac{c^3 kA}{4\hbar G}$$

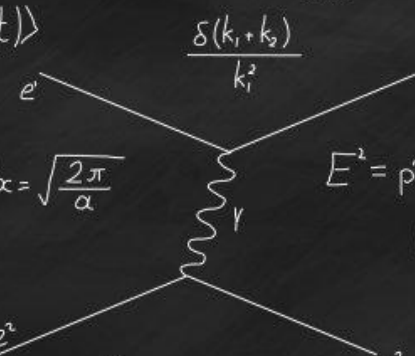
$$L = tr\left\{ \frac{1}{g^2} F_{IJ} F^{IJ} - i\lambda \Gamma^I D_I \lambda \right\}$$

$$H|\psi(t)\rangle = i\hbar \frac{\partial}{\partial t} |\psi(t)\rangle$$

$$E = mc^2$$

$$E^2 = (pc)^2 + (mc^2)^2$$

$$r = \frac{\theta}{2\pi} + \frac{4\pi}{g^2}$$

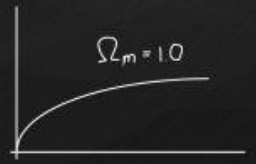


$$I = \int e^{-ax^2/2} dx = \sqrt{\frac{2\pi}{a}}$$

$$E^2 = p^2 c^2 + m^2 c^4$$

$$\frac{1}{c^2} \frac{\partial^2}{\partial t^2} \psi - \nabla^2 \psi + \frac{m^2 c^2}{\hbar^2} \psi = 0$$

$$p = \hbar k = \frac{h\nu}{c} = \frac{h}{\lambda}$$



$$S = \frac{1}{2} \int d^4x \left(R + \frac{R^2}{6M^2} \right)$$

$$A_{ij} = \frac{8\pi h \nu^3}{c^3} B_{ij}$$

$$S_{fi} = \langle f | S | i \rangle$$

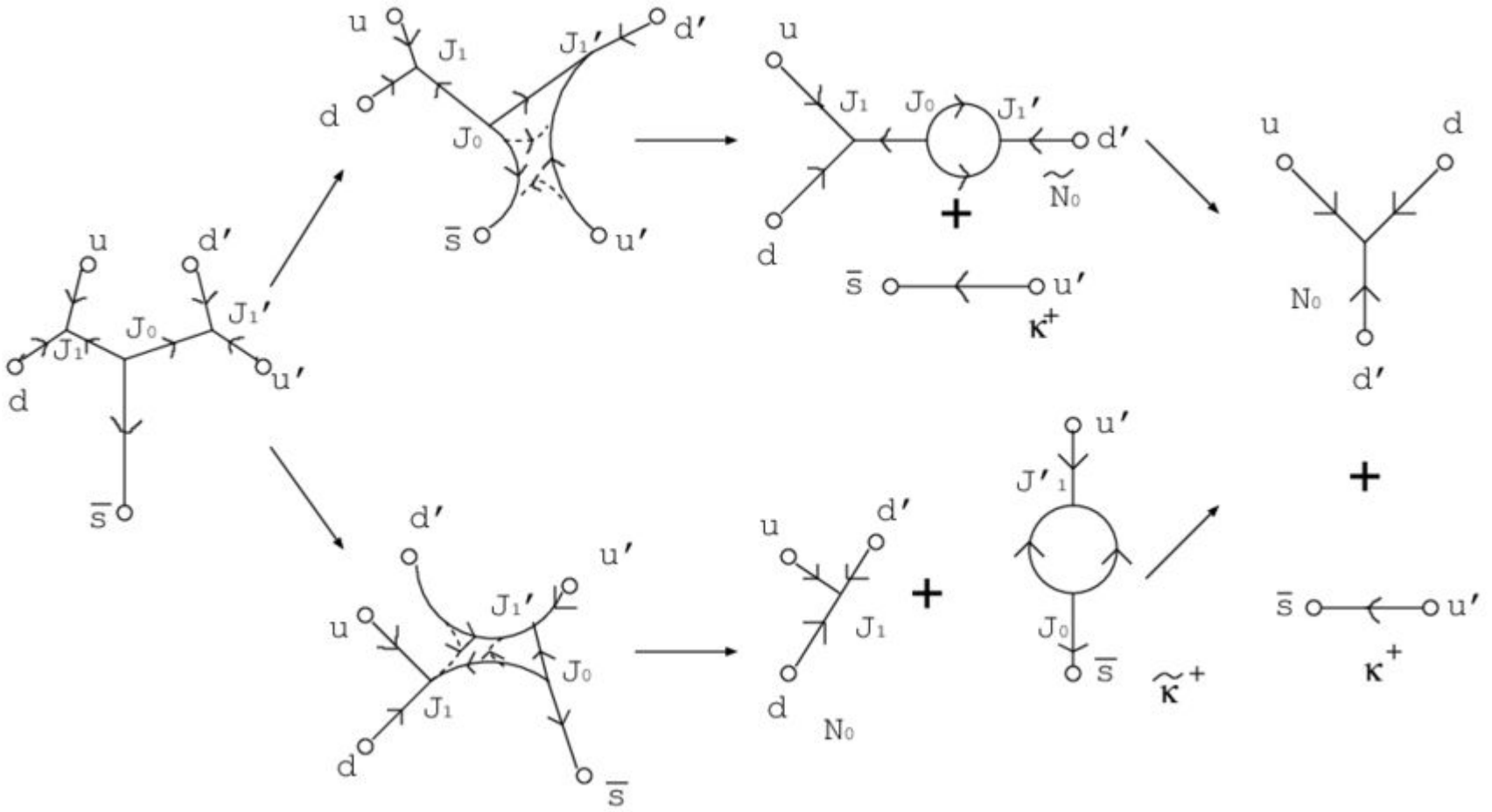
$$dY = e^{-\int_t^s V(X_{r,r}) dr_\Theta (X,s)} \frac{\partial u}{\partial X} dW$$

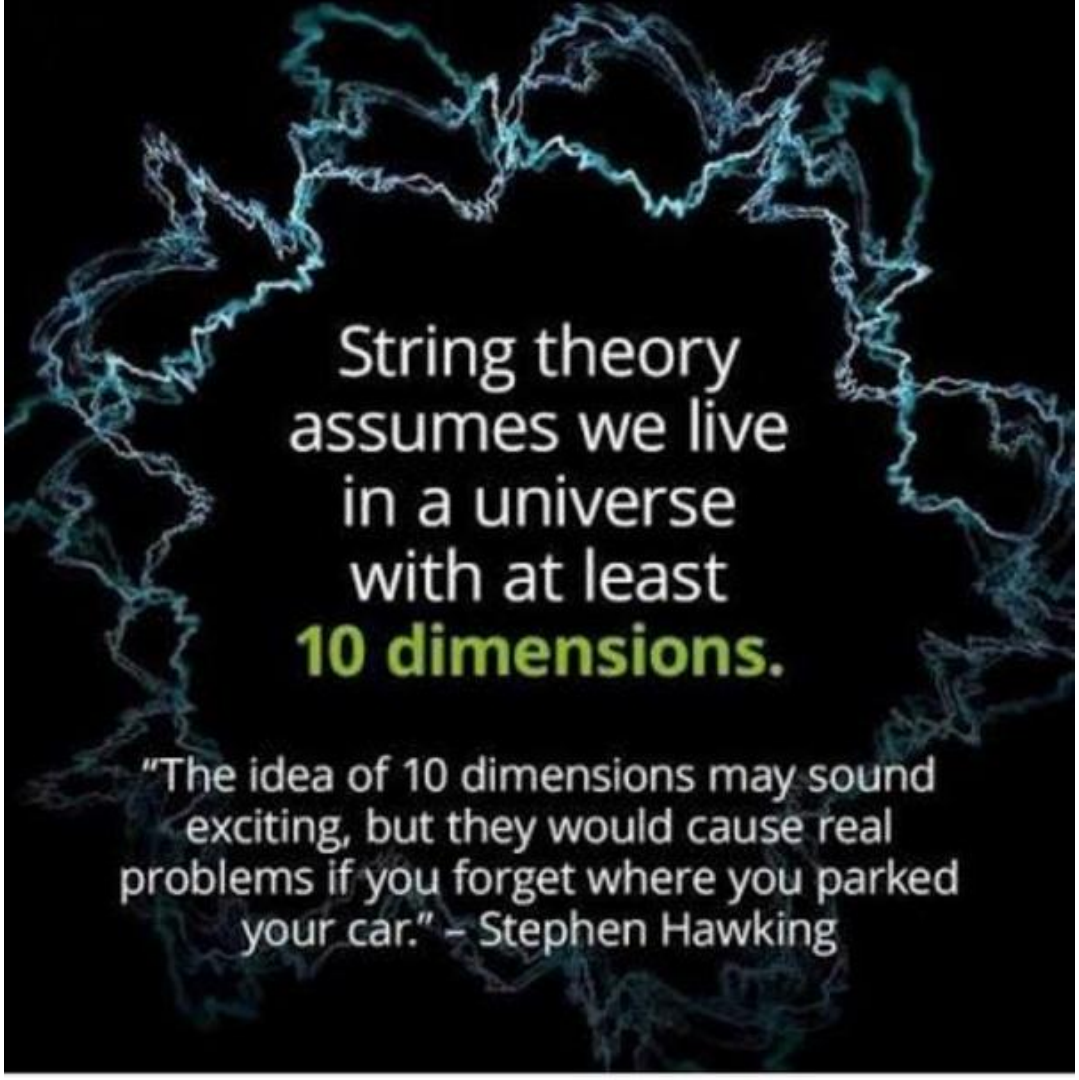
$$\frac{d}{dt} \langle A \rangle = \frac{1}{i\hbar} \langle [\hat{A}, \hat{H}] \rangle + \left\langle \frac{\partial \hat{A}}{\partial t} \right\rangle$$

$$i\hbar \frac{\partial}{\partial t} \psi = -\frac{\hbar^2}{2} \sum_{n=1}^{N} \frac{1}{m_n} \nabla_n^2 \psi + V\psi$$

$$\Delta x \Delta p \geq \frac{\hbar}{2}$$


STRING THEORY RESEARCH


u
d
s̄
d′
u′
J₀
J₁
J₁′
N₀
κ⁺
J′₁
κ̃⁺
+


String theory
assumes we live
in a universe
with at least
10 dimensions.
"The idea of 10 dimensions may sound
exciting, but they would cause real
problems if you forget where you parked
your car." – Stephen Hawking

Types (so far)

- int
- float
- str

Some basic string theory

- Use ‘single quote’ or “double quote” or “‘triple quotes’”
- Python strings are immutable
- Strings are not necessarily one character per byte - they are Unicode
- Strings may contain “escape” characters with backslash: \n, \”,...
- Determine length via function `len()`
- String methods: `upper()`, `lower()`, `replace()`, `find()`, `startswith()`, `split()`, `strip()`...
- Use + to concatenate strings
- String interpolation
- String slicing

String interpolation

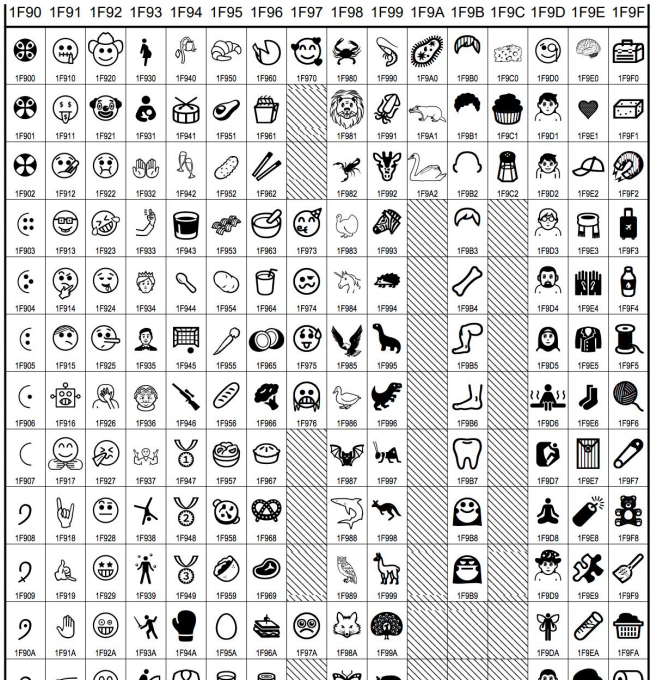
1. In functions like print(), include arguments as comma separated values.
   a. print(‘the value of’, ‘myvar’, ‘is’, a)
2. Classic method: using % entities in a string, followed by % and a list
   a. ‘The value of %s is %s’ % (‘myvar’, a)
3. Using .format()
   a. ‘The value of {} is {}’.format(‘myvar’, a)
   b. ‘The value of {0} is {1}’.format(‘myvar’, a)
   c. ‘The value of {varname} is {varval}’.format(varname=’myvar’, varval=a)
4. f-strings (Python 3.6 and later)
   a. f’The value of {“myvar”} is {a}’
5. Template strings

Unicode

- Strings are arrays of Unicode characters (“code points”)
- Unicode code points: 0 through 0x10FFFF (1,114,112) (21 bits)
- *Encoded* into smaller *bytes* for efficient transmission and storage
- Common encoding: utf-8
- ASCII characters (< 128) are valid utf-8 characters (1 byte)
- Other characters in utf-8 can take 2, 3 or 4 bytes
- **Always use Unicode internally. *Decode* what you receive, and *encode* what you send**.

Unicode character codes examples

0x1f921
0x4e7e



Unicode examples

- ℙƴ☂ℌøἤ
- ȧƈƈḗƞŧḗḓ ŧḗẋŧ ƒǿř ŧḗşŧīƞɠ
- ¡ooʇ ןnɟǝsn sı uʍop-ǝpısdn
- Hello, World!

String Slicing

s = ‘abcd’
s[0]
s[-1]
s[2:]
s[:]
s[1:2]
s[0:100]
s[1:-1]
s[-2:]
s[:n] + s[n:] is always a

Functions can be Fun

Function domains

- Built-ins
- User-defined in current file
- User-defined in another file (module)
- Standard library (module)

Fun Function Facts

- Function is a type (like int, float, str...)
- Functions may contain parameters (arguments), passed in
- Functions may return a value, or multiple values
- Arguments and return values can be any type, including function
- A function can call other functions, (including itself)
- A function may contain definitions of new functions
- Scope of variables inside a function are local to that function, and invisible from outside the function
- Functions may expect a specific number of arguments, or allow a variable number of arguments
- The value of some function arguments may have defaults
- Object references are passed by value.