MathsJam in Python 💻

by Stefania Delprete @astrastefania

I graduated in Physics

My final score was 108 and it was April 27 and...

I graduated in Physics

My final score was 108 and it was April 27 and...

$$108 = 4 * 27 = 1^1 * 2^2 * 3^3$$

🔔 108 is the **hyperfactorial of 3** 🎉 🎉

I graduated in Physics

My final score was 108 and it was April 27 and...

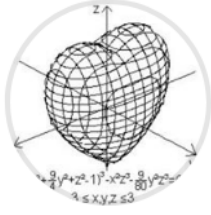
$$108 = 4 * 27 = 1^1 * 2^2 * 3^3$$

🔔 108 is the **hyperfactorial of 3** 🎉 🎉

My family, friends and even the local LUG didn't get my enthusiam for such number...

Moved to London, UK

Started mathdate page...



mathdate

1 min ·

Today is 18/11/2018,
and 18 + 1 + 1 - 2 = 0! * 18.

What's your #mathdate? 🤔

Discovered the MathsJam! 🎲 📝

(and the discussion between *Math* and *Maths*)

Moved to Berlin, Germany 🇩🇪

MathsJam Berlin @BerlinMathsJam

and Python made its appearance...

Oh, by the way Python is from Monthy Python, not the 🐍

Moved back to Turin, Italy 🇮🇹

MathsJam Turin @TRNMathsJam

"Resistance is useless... "

So on August 21, MathsJam Shout from Bristol

Make Twenty-one

Make 21 out of the numbers **1, 5, 6** and **7** using only basic arithmetic with the operations: + − × ÷ and using each number exactly once. You can use brackets, but you can't make up numbers like 15 from 1 and 5.

and after a while trying...

... we started thinking how to code the solution 🔒

In [11]:

```python
# Code by Matteo Bertini

import itertools

# Permutation of the four numbers
for a,b,c,d in itertools.permutations([1,5,6,7]):
    # Inserting operators
    for op1, op2, op3 in itertools.product("+-*/", "+-*/", "+-*/"):
        # Inserting the brackets
        for prec in (f"{a} {op1} ({b} {op2} ({c} {op3} {d}))",
                     f"({a} {op1} {b}) {op2} ({c} {op3} {d})",
                     f"(({a} {op1} {b}) {op2} {c}) {op3} {d}"):
            print(prec)
```

```
1 + (5 + (6 + 7))
(1 + 5) + (6 + 7)
((1 + 5) + 6) + 7
1 + (5 + (6 - 7))
(1 + 5) + (6 - 7)
((1 + 5) + 6) - 7
1 + (5 + (6 * 7))
(1 + 5) + (6 * 7)
((1 + 5) + 6) * 7
1 + (5 + (6 / 7))
(1 + 5) + (6 / 7)
((1 + 5) + 6) / 7
1 + (5 - (6 + 7))
(1 + 5) - (6 + 7)
((1 + 5) - 6) + 7
1 + (5 - (6 - 7))
```

...and the morning after 🔑

In [12]:
```python
# Code by Matteo Bertini

import itertools

# Permutation of the four numbers
for a,b,c,d in itertools.permutations([1,5,6,7]):
    # Inserting operators
    for op1, op2, op3 in itertools.product("+-*/", "+-*/", "+-*/"):
        # Inserting the brackets
        for prec in (f"{a} {op1} ({b} {op2} ({c} {op3} {d}))",
                     f"({a} {op1} {b}) {op2} ({c} {op3} {d})",
                     f"(({a} {op1} {b}) {op2} {c}) {op3} {d}"):
            # print(prec)

            # Evaluating all the possible operations
            try:
                res = eval(prec)
            except ZeroDivisionError:
                res = 0
            if abs(res) == 21:
                print(f"{prec} = {int(res)}")
```

```
6 / (1 - (5 / 7)) = 21
```

😉 So yes, the solution is

$$\frac{6}{1-\frac{5}{7}} = 21$$

Thanks, love you all! ❤️

I'm having so much fun 😄

Stefania Delprete @astrastefania

PS: slides made in markdown, Python script run in Jupyter Notebook