

1 Sixth principle of the Zen of Python explained and applied.

"Spacing is better than dense," is the sixth principle of the Zen of Python, which indicates that the code we have should not be tightly packed so that variables can be recognized more easily by any developer's glance.

1.1 Example in C++

```
//Aplicacion correcta del principio del codigo en C++
//Codigo de que genera aleatorios entre 20 y 80 y resuelve un operacion
#include <iostream>
#include <cstdlib>
#include <ctime>
#include <cmath>

#define PI 3.141592 // Definicion del valor de PI

using namespace std;

int main() {
    int max, min;
    printf("\nIngrese valor <max>:-");
    scanf("%d", &max);
    printf("Ingrese valor <min>:-");
    scanf("%d", &min);

    srand(time(NULL));
    int ale = rand() % (max - min + 1) + min;
    float rad = ale * (PI / 180), Fx = 0;

    if (80 > ale && ale > 50) {
        // Calcula numero * (sin^2(num))
        Fx = ale * pow(sin(rad), 2);
        printf("\nNumero generado: %d\n", ale);
        printf("num * (sin^2(rad)): %.2f\n", Fx);
    } else if (20 < ale && ale < 40) {
        // Calcula (2 * cos(num)) / num
        if (ale == 0) {
            cout << "No se puede dividir por cero." << endl;
        } else {
            Fx = (2 * cos(rad)) / ale;
            printf("\nNumero generado: %d\n", ale);
            printf("(2 * cos(rad)) / num: %.2f\n", Fx);
        }
    } else {
        // Calcula ((3 * tan(num)) - (2 * sin(num))) / (num + 2)
        if (ale == -2) {
            cout << "No se puede dividir por -2." << endl;
        } else {
            Fx = ((3 * tan(rad)) - (2 * sin(rad))) / (ale + 2);
            printf("\nNumero generado: %d\n", ale);
            printf("((3 * tan(rad)) - (2 * sin(rad))) / (num + 2): %.2f\n", Fx);
        }
    }

    return 0;
}
```

1.2 Example y Python

```
# Aplicacion correcta del principio del codigo en python
# Programa que calcula el factorial de n
import math

def factorial(n):
    if n == 0:
        return 1
    else:
        return math.factorial(n)
```

```
def main():
    A = 1
    while A != 0:
        A = int(input("Ingrese <A>: "))
        if A == 0:
            break
        V = int(input("Ingrese <V>: "))
        for i in range(1, V + 1):
            n = int(input("Ingrese <n>: "))
            fac = factorial(n)
            print("El factorial de <%d> es: %d" % (n, fac))

    if __name__ == "__main__":
        main()
```