

Trabajo final de curso de Fundamentos de la programación 2024-I

Delgado, G. (60 %)

Escuela de Ciencias de la Computación
Universidad Nacional de Ingeniería
Lima, Perú
gustavo.delgado.r@uni.pe

Mori, J. (40 %)

Escuela de Ciencias de la Computación
Universidad Nacional de Ingeniería
Lima, Perú
jean.mori.m@uni.pe

Resumen—El curso de fundamentos de programación ofrecido por la Universidad Nacional de Ingeniería aborda principalmente la enseñanza de algunos de los conceptos base de programación en el lenguaje C++, como la manipulación de archivos binarios, el uso de punteros o la manipulación de memoria dinámica. Sin embargo, como proyecto final del curso, la plana docente decidió plantear una serie de problemas desde muy básicos a algunos con cierta complejidad utilizando el lenguaje de programación python.

Así pues, el presente documento tiene como finalidad exponer los conceptos investigados para la resolución de los problemas propuestos por los profesores del curso, así como también dar a conocer los resultados del desarrollo de dichos problemas.

Índice de términos— Factorial, Máximo común divisor, Clases en Python, Machine Learning.

I. INTRODUCCIÓN

Python es un lenguaje de programación versátil y poderoso ampliamente usado en varios campos, incluyendo la ciencia de datos, desarrollo web y Computación científica. Uno de los aspectos por los que destaca python es por su simplicidad y su fácil lectura. Aquí hacemos una simple comparación de códigos entre python y c++:

Python:

```
1 print("Hola mundo")
```

Listing 1. Script de bienvenida en Python

C++:

```
1 #include <iostream>
2 using namespace std;
3 int main() {
4     cout<<"Hola mundo";
5     return 0;
6 }
```

Listing 2. Script de bienvenida en C++

II. MARCO TEÓRICO

II-A. Función Factorial

La función factorial calcula el producto de todos los enteros positivos desde 1 hasta un número entero no negativo dado n . Se denota como $n!$.

$$n! = n \cdot (n-1) \cdot (n-2) \cdot \dots \cdot 2 \cdot 1$$

Por ejemplo:

- $5! = 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1 = 120$
- $0! = 1$ (por convención)

II-A1. Definición Recursiva: La función factorial se puede definir recursivamente de la siguiente manera:

$$n! = \begin{cases} 1 & \text{si } n = 0 \\ n \cdot (n-1)! & \text{si } n > 0 \end{cases}$$

II-A2. Propiedades:

- $n! > 0$ para todos los enteros no negativos n .
- $0! = 1$ (por convención).
- El factorial crece rápidamente: $n! > n^2$ para n suficientemente grande.

II-B. Algoritmo de Euclides

El algoritmo de Euclides se utiliza para encontrar el ****máximo común divisor (MCD)**** de dos enteros positivos. Dados dos números a y b , el algoritmo procede de la siguiente manera:

1. Realiza divisiones sucesivas:

$$a \bmod b = R$$

donde R es el residuo al dividir a entre b .

2. Actualiza los valores:

$$a = b \quad y \quad b = R$$

Repite los pasos 1 y 2 hasta que $a \bmod b$ sea mayor que 0.

3. El MCD es igual al valor final de b .

II-C. *Clases en Python*

En Python, las **clases** son como **constructores de objetos** o **plantillas** para crear instancias. Cada objeto creado a partir de una clase tiene sus propias **propiedades** (variables) y **métodos** (funciones).

II-C1. *title:*

III. ESTADO DEL ARTE

III-A. *Transformador Universal*

III-B. *BERT*

III-C. *GPT-2*

III-C1. *Resultados:*

III-D. *Transformadores dispersos*

III-E. *XLNet*

IV. METODOLOGÍA

V. EXPERIMENTACIÓN Y RESULTADOS

VI. CONCLUSIONES

VII. DISCUSIÓN

VIII. CONCLUSIONES Y TRABAJOS FUTUROS

REFERENCIAS

- [1] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer. Deep contextualized word representations. NAACL 2018.
- [2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding (2018). Disponible en <https://arxiv.org/abs/1810.04805>.