

**UNIVERSIDAD MAYOR DE SAN ANDRÉS**  
**FACULTAD DE CIENCIAS PURAS Y NATURALES**  
**CARRERA DE INFORMÁTICA**



---

**PRIMER PARCIAL**  
**SISTEMAS EN TIEMPO REAL Y DISTRIBUIDO**

---

**SIGLA:**

INF-317

**NOMBRE:**

Richard Pomacosi Quispe

**C.I.:**

9994735 LP.

**PARALELO:**

A

**DOCENTE:**

Lic. Silva Choque Moises Martin

**GESTION:**

I/2020

La Paz – Bolivia

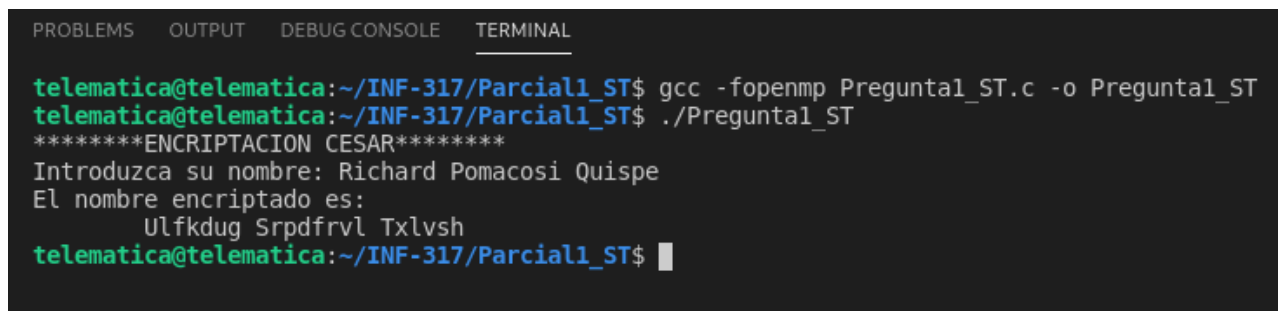
## 1. En OPENMP cree un programa que encripte su nombre mediante CESAR

OpenMP es API multiplataforma para multiprocesos de memoria compartida. Está escrito para usarse de manera nativa desde C, C++ y Fortran. Se encuentra disponible para una gran variedad de arquitecturas y sistemas operativos, entre los que podemos destacar Windows, Mac OS X y la familia de sistemas operativos Unix. OpenMP es una estándar y está bajo la administración de un consorcio no lucrativo impulsado por los mayores productores de hardware.

Para la compilación se utilizó el siguiente comando (Debian-GNU/Linux):

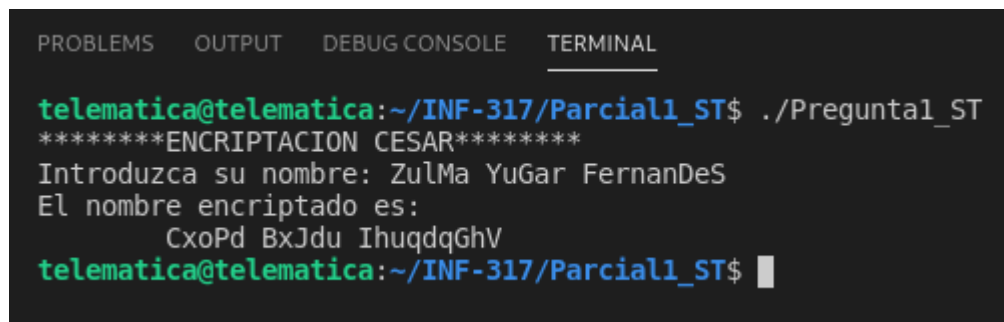
```
gcc -fopenmp Pregunta1_ST.c -o Pregunta1_ST
```

A continuación se muestra el programa desarrollado C utilizando el estandar OpenMP, El programa solicita como entrada una cadena (nombre) y posteriormente este lo encripta mediante CESAR y lo muestra en pantalla.



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
telematica@telematica:~/INF-317/Parcial1_ST$ gcc -fopenmp Pregunta1_ST.c -o Pregunta1_ST
telematica@telematica:~/INF-317/Parcial1_ST$ ./Pregunta1_ST
*****ENCRIPACION CESAR*****
Introduzca su nombre: Richard Pomacosi Quispe
El nombre encriptado es:
    Ulfkdug Srpdfrvl Txlvsh
telematica@telematica:~/INF-317/Parcial1_ST$
```

Otros ejemplos:



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
telematica@telematica:~/INF-317/Parcial1_ST$ ./Pregunta1_ST
*****ENCRIPACION CESAR*****
Introduzca su nombre: ZulMa YuGar FernanDeS
El nombre encriptado es:
    CxoPd BxJdu IhuqdgGhV
telematica@telematica:~/INF-317/Parcial1_ST$
```

## 2. En Phyton(no paralelo) cree un programa que encripte su nombre mediante CESAR

Python es un lenguaje de programación interpretado de tipado dinámico cuya filosofía hace hincapié en una sintaxis que favorezca un código legible. Se trata de un lenguaje de programación multiparadigma y disponible en varias plataformas.

El cifrado César es uno de los primeros métodos de cifrado conocidos históricamente. Julio César lo usó para enviar órdenes a sus generales en los campos de batalla. Consistía en escribir el mensaje con un alfabeto que estaba

formado por las letras del alfabeto latino normal desplazadas tres posiciones a la derecha. Con nuestro alfabeto el sistema quedaría así:

<b>Alfabeto:</b>	A B C D E F G H I J K L M N Ñ O P Q R S T U V W X Y Z
<b>Alfabeto cifrado:</b>	D E F G H I J K L M N Ñ O P Q R S T U V W X Y Z A B C

A continuación, se muestra las respectivas capturas de pantalla:

The screenshot shows a Python script named 'pregunta2.py' and its execution in a terminal. The script implements a Caesar cipher with a shift of 3. It processes the input string 'Richard Pomacosi Quispe' and outputs the encrypted string 'Ulfkdug Srpdfrvl Txlvsh'. The terminal output is as follows:

```
PS D:\Informatica\7_Sistemas_en_tiempo_Real_y_D_INF-317\Parcial1_ST> python pregunta2.py
*****ENCRIPCIÓN CESAR*****
Introduce tu nombre: Richard Pomacosi Quispe
Su nombre encriptado es:
Ulfkdug Srpdfrvl Txlvsh
PS D:\Informatica\7_Sistemas_en_tiempo_Real_y_D_INF-317\Parcial1_ST>
```

Como se indica en el enunciado se encripto el nombre de mi persona esto es:

**Texto: Richard:** Pomacosi Quispe

**Texto Encriptado:** Ulfkdug Srpdfrvl Txlvsh

Se probó también con otros datos como se muestra en la siguiente captura:

```
PS D:\Informatica\7_Sistemas_en_tiempo_Real_y_D_INF-317\Parcial1_ST> python pregunta2.py
*****ENCRIPCIÓN CESAR*****
Introduce tu nombre: Zulma YuGar FerNandEs
Su nombre encriptado es:
    Cxopd BxJdu IhuQdqgHv
PS D:\Informatica\7_Sistemas_en_tiempo_Real_y_D_INF-317\Parcial1_ST> █
```

**Texto:** Zulma YuGar FerNandEs

**Texto Encriptado:** Cxopd BxJdu IhuQdqgHv

### 3. En OPENMP realice una solución de matrices mediante JACOBI

Luego de desarrollar el programa se evaluó con dos ecuaciones los cuales son:

Ecuación 1:

$$4X_1 + X_2 - X_3 = 5$$

$$-X_1 + 3X_2 + X_3 = -4$$

$$2X_1 + 2X_2 + 5X_3 = 1$$

La forma matricial de esta ecuación es: **Ax=b**

Se ejecutó el siguiente comando:

```
gcc -fopenmp Pregunta3_ST.c -o Pregunta3_ST
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
telematica@telematica:~/INF-317/Parcial1_ST$ gcc -fopenmp Pregunta3_ST.c -o Pregunta3_ST
telematica@telematica:~/INF-317/Parcial1_ST$ ./Pregunta3_ST *****
* PROGRAMACION CON OMP *
*****
```

Se introdujeron los datos según la indicación del programa.

```
telematica@telematica:~/INF-317/Pa
telematica@telematica:~/INF-317/Pa
* PROGRAMACION CON OMP *
* METODO DE JACOBI *
* Ax=b *
*****
Numero de variables: 3
Ingrese Matriz A
Posicion(0,0)=4
Posicion(0,1)=1
Posicion(0,2)=-1
Posicion(1,0)=-1
Posicion(1,1)=3
Posicion(1,2)=1
Posicion(2,0)=2
Posicion(2,1)=2
Posicion(2,2)=5
Ingrese Matriz b
Posicion(0,0)=5
Posicion(1,0)=-4
Posicion(2,0)=1
MATRIZ A
```

Se muestra las matrices  $Ax=b$  y las iteraciones del método Jacobi

```

Posicion(2,0)=1
MATRIZ A
  4.000    1.000   -1.000
 -1.000    3.000    1.000
  2.000    2.000    5.000
Matriz B
  5.000
 -4.000
  1.000
Iteraciones:
  0.000    0.000    0.000
  1.250   -1.333    0.200
  1.633   -0.983    0.233
  1.554   -0.867   -0.060
  1.452   -0.795   -0.075
  1.430   -0.824   -0.063
  1.440   -0.836   -0.042
  1.448   -0.839   -0.042
  1.449   -0.837   -0.044
  1.448   -0.836   -0.045
  1.448   -0.836   -0.045
  1.448   -0.836   -0.045
  1.448   -0.836   -0.045
  1.448   -0.836   -0.045

```

La ultima iteración representará la solución de la ecuación esto es:

```

  1.448   -0.836   -0.045
  1.448   -0.836   -0.045
  1.448   -0.836   -0.045
  1.448   -0.836   -0.045
  1.448   -0.836   -0.045
Resultado:
X1 -->  1.448
X2 --> -0.836
X3 --> -0.045
telematica@telematica:~/INF-317/Parcial1_ST$

```

Comprobamos estos resultados con los obtenidos en un hoja EXCEL, donde se comprueba el resultado.

A				B	
4	1	-	1	5	
-	1	3	1	-4	
	2	2	5	1	
A^-1				Solución	(A^-1)*B
0,194030	-0,104478	0,059701			1,447761
0,104478	0,328358	-0,044776			-0,835821
-0,119403	-0,089552	0,194030			-0,044776

Ecuación 2:

$$3X_1 - X_2 - X_3 = 1$$

$$-X_1 + 3X_2 + X_3 = 0$$

$$2X_1 + 2X_2 + 5X_3 = 4$$

Se muestra el resultado calculado por el programa:

```

0.035    -0.237    0.658
Resultado:
X1  -->  0.035
X2  --> -0.237
X3  -->  0.658
telematica@telematica:~/INF-317/Parcial1_ST$
Ln 50, Col 19  Spaces: 4  UTF

```

Al igual que en el anterior ejercicio se comprobó los resultados

A				B	
3	-	1	1	1	
3		6	2	0	
3		3	7	4	
A <sup>-1</sup>				Solución	(A <sup>-1</sup> )*B
0,315789	0,087719	-0,070175			0,035088
-0,131579	0,157895	-0,026316			-0,236842
-0,078947	-0,105263	0,184211			0,657895