

**UNIVERSIDAD NACIONAL DE TRUJILLO**

Facultad de Ingeniería

Programa de Ingeniería Mecatrónica

LABORATORIO N° 2

“ALGORITMIA CONDICIONAL”

**DESARROLLO DE GUIA DE LABORATORIO**

PROGRAMACIÓN I

**ESTUDIANTE(S) :**

**SÁNCHEZ ROJAS JHONATAN ARTEMIO**

**VALDIVIEZO JIMÉNEZ VÍCTOR JAVIER**

**VIGO VILLAR CRISTHIAN AARON**

**DOCENTE :**

**ASTO RODRIGUEZ EMERSON MAXIMO**

**CICLO :**

**2022 I**

Trujillo, Perú

2022

**INDICE**

[RESUMEN 3](#_Toc105768910)

[DESARROLLO DEL LABORATORIO 4](#_Toc105768911)

[1.1. Desarrollo de la experiencia 4](#_Toc105768912)

[a) Ejercicio 1 4](#_Toc105768913)

[b) Ejercicio 2 6](#_Toc105768914)

[c) Ejercicio 3 7](#_Toc105768915)

[d) Ejercicio 4 8](#_Toc105768916)

[1.2. Resultados de la experiencia 9](#_Toc105768917)

[a) Ejercicio 1 9](#_Toc105768918)

[b) Ejercicio 2 9](#_Toc105768919)

[c) Ejercicio 3 10](#_Toc105768920)

[d) Ejercicio 4 11](#_Toc105768921)

[1.3. Desarrollo de test de comprobación 12](#_Toc105768922)

[1.4. Recomendaciones 13](#_Toc105768923)

[1.5. Conclusiones 13](#_Toc105768924)

[REFERENCIAS BIBLIOGRÁFICAS 14](#_Toc105768925)

[ANEXOS 15](#_Toc105768926)

# RESUMEN

La presentación del siguiente laboratorio supone la elaboración de seudocódigos a través del software PSeInt en solución a 4 problemas específicos propuestos sobre algoritmia condicional.

Durante el desarrollo se pudo analizar cada problema de manera detallada; para así, llegar a su correcta presentación en forma de sentencias condicionales, haciendo la respectiva comprobación ejecutando el seudocódigo y dándole valores al azar. Las respuestas derivadas del programa fueron comprobadas manualmente para llegar a una correcta verificación.

Finalmente, ayudándonos de la misma herramienta PSeInt, se pudo llegar a los diagramas de flujo de cada seudocódigo creado, los mismos que son expuestos en la parte -anexos- del presente documento.

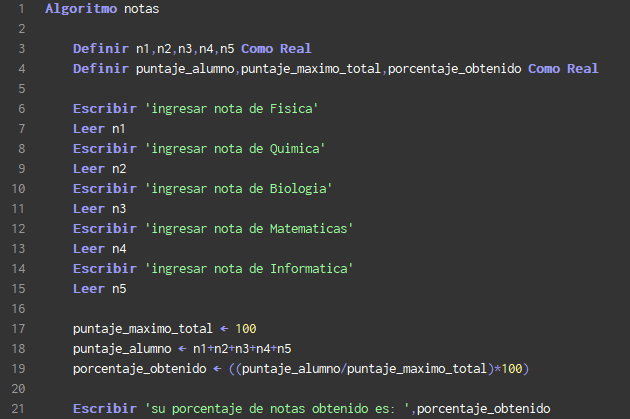
# DESARROLLO DEL LABORATORIO

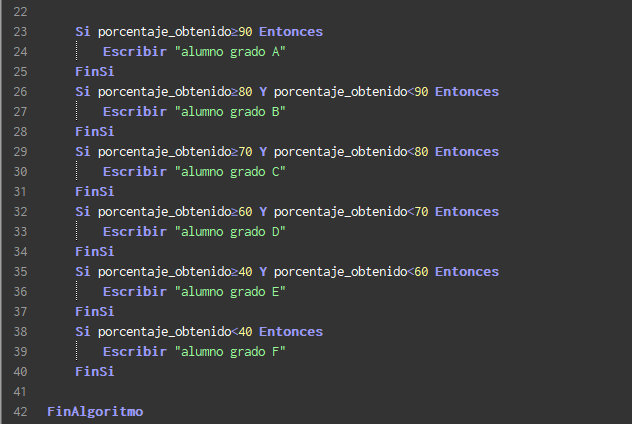
## Desarrollo de la experiencia

### Ejercicio 1

Valdiviezo Jimenez, Victor Javier

* Escriba un programa para ingresar calificaciones de cinco materias: Física, Química, Biología, Matemáticas e Informática. Calcule el porcentaje y el grado de acuerdo con lo siguiente:
  + Porcentaje> = 90%: Grado A
  + Porcentaje > = 80%: Grado B
  + Porcentaje> = 70%: Grado C
  + Porcentaje> = 60%: Grado D
  + Porcentaje> = 40%: Grado E
  + Porcentaje <40%: Grado F



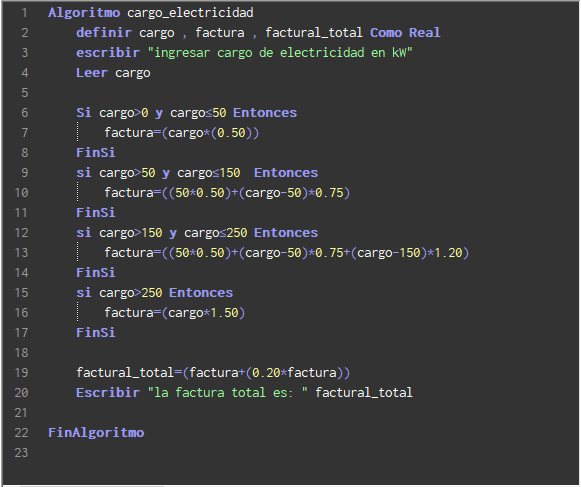


### Ejercicio 2

Valdiviezo Jimenez, Victor Javier

* Escriba un programa para ingresar los cargos por unidad de electricidad y calcular la factura total de electricidad de acuerdo con la condición dada:
  + Para los primeros 50 Kw. 0,50/Kw
  + Para los próximos 100 Kw. 0,75/kw
  + Para los próximos 100 unidades Kw. 1,20/Kw
  + Para consumos por encima de 250 Kw. 1,50/Kw

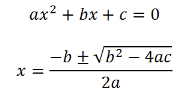
Se añade un recargo adicional del 20% a la factura por otros gastos.

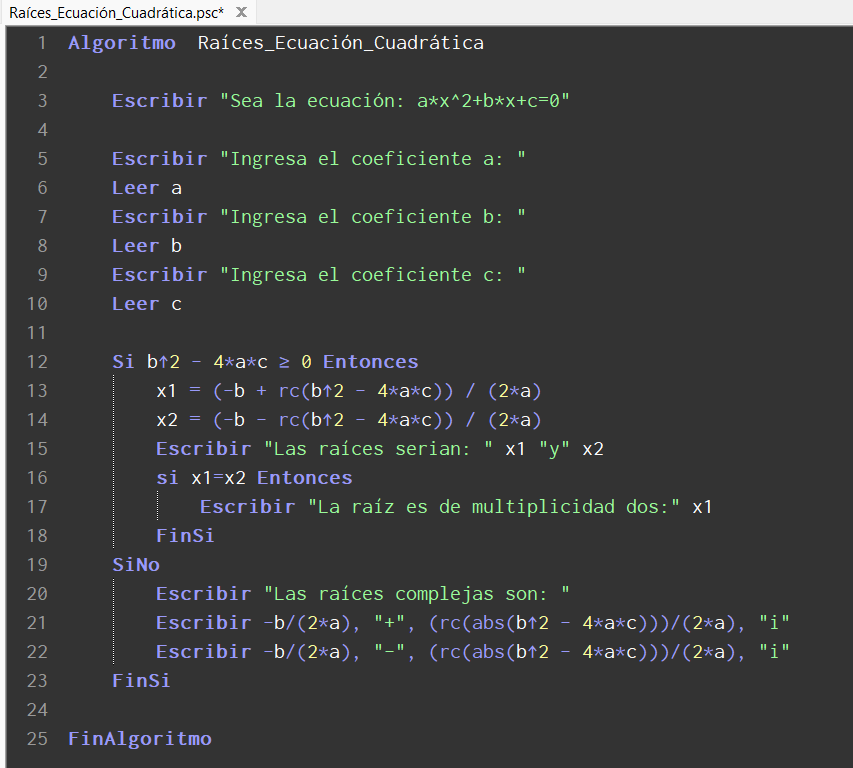


### Ejercicio 3

Sánchez Rojas, Jhonatan Artemio

Escriba un programa para encontrar las raíces de una ecuación cuadrática.

****

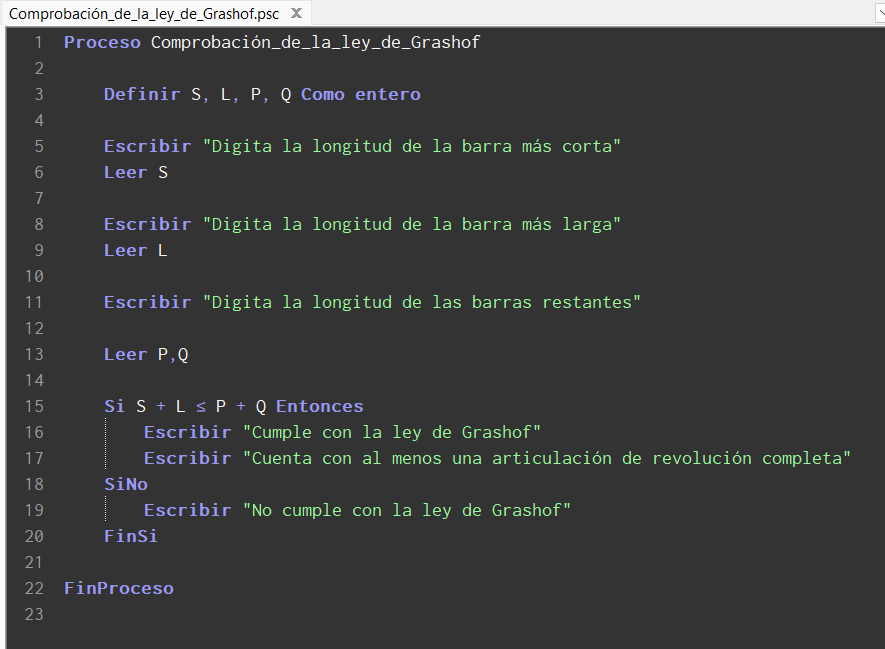
****

### Ejercicio 4

Vigo Villar, Cristhian Aaron

La Ley de Grashof establece que un mecanismo de cuatro barras tiene al menos una articulación de revolución completa, si y solo si la suma de las longitudes de la barra más corta (S) y la barra más larga (L) es menor o igual que la suma de las

longitudes de las barras restantes (P y Q). Escriba un programa que solicite las longitudes de las 4 barras de un mecanismo y pueda determinar si cumple con la ley de Grashof.

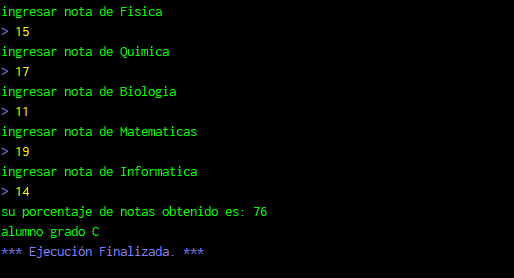


## Resultados de la experiencia

Link del repositorio: <https://github.com/hdjdudixjxy/Laboratorio-2-Grupo1-PI-UNT-2022.git>

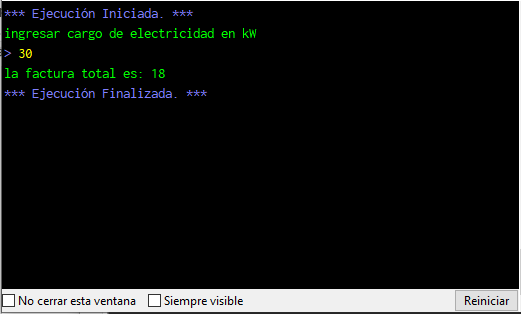
### Ejercicio 1

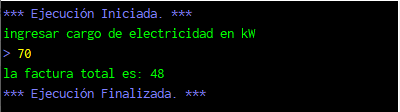
El resultado obtenido fue:

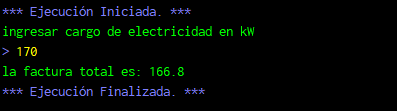


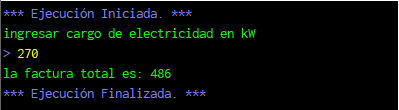
### Ejercicio 2

El resultado obtenido fue:



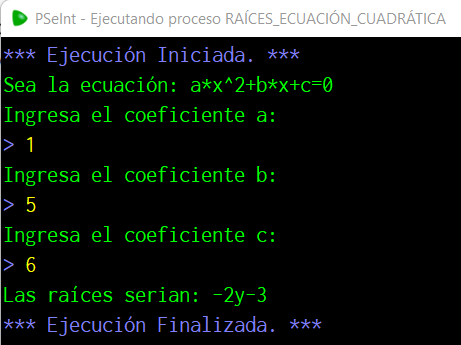


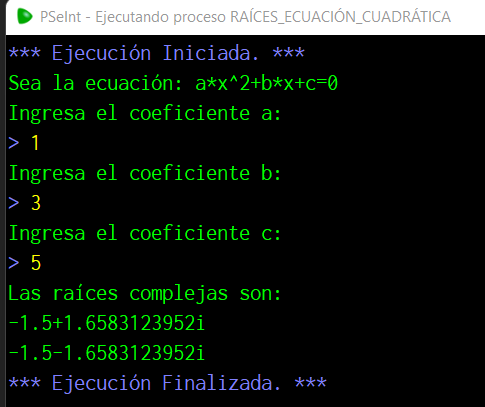




### Ejercicio 3

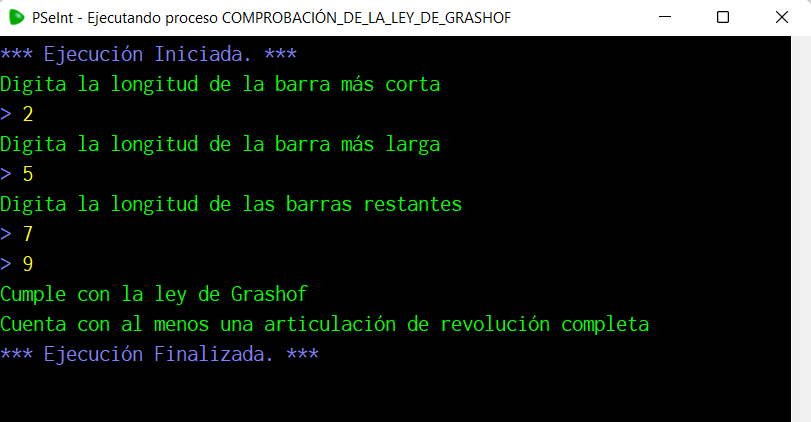
El resultado obtenido fue:



****

### Ejercicio 4

El resultado obtenido fue:

****

### 

## Desarrollo de test de comprobación

a) Investigue que otros comandos de git se pueden utilizar para crear una nueva rama.

A parte del conocido “git branch” para la creación de ramas, existe otro comando con el cual podemos crear una nueva rama a partir de una ya existente. Este comando se trabaja de la siguiente manera: ***git checkout –b “nombre de rama”***. Y supone una manera alternativa al “git branch” para algunos casos específicos.

b) Explique la sentencia Si… entonces.

Este comando permite colocar condiciones al algoritmo, al aplicarlo, se evalúa la condición y se ejecuta las instrucciones que correspondan.

Ejm:

Si x es impar Entonces

escribir " pertenece al conjunto A"

c) Explique la sentencia según.

Este comando permite ejecutar acciones según únicamente el valor de una variable, que ha diferencia de “Si” puede llevar una condición para la variable y “Según” no.

## Recomendaciones

* Ser ordenado al momento de colocar las sentencias.
* Al momento de hacer el pseudocódigo, utilizar términos que sean fáciles de reconocer en el futuro.
* Pensar en la solución mas sencilla antes de realizar el pseudocódigo.

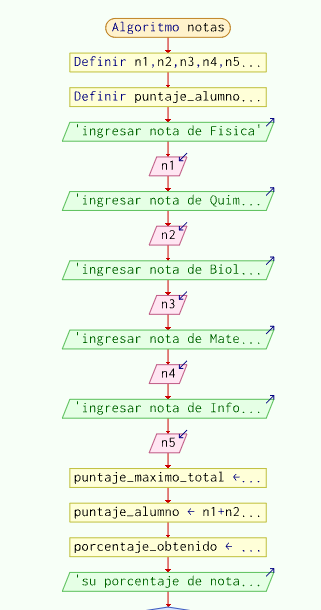
## Conclusiones

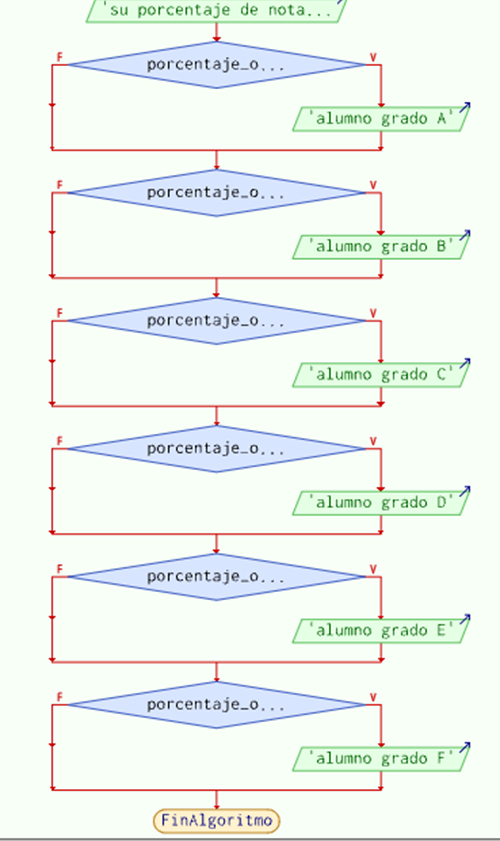
* Los ejercicios fueron desarrollados de manera eficaz, haciendo uso de sentencias condicionales, lo cual era el objetivo principal de este informe de laboratorio.
* Las sentencias condicionales permiten crear algoritmos más precisos en cuanto a la respuesta, dado que, permite establecer ciertos parámetros.

# REFERENCIAS BIBLIOGRÁFICAS

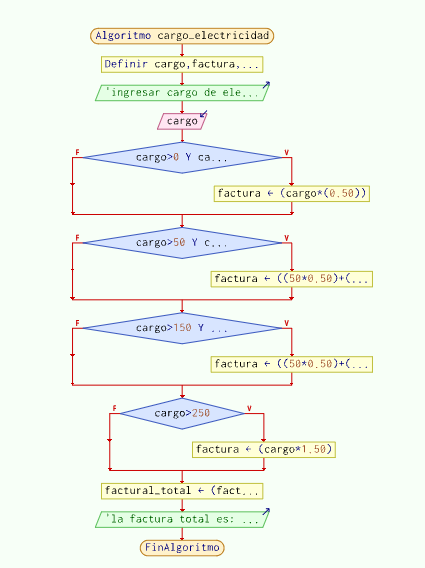
Delgado, P. (2014). *Algoritmos resueltos con diagramas de flujo y pseudocódigo.* México: Universidad Autónoma de Aguascalientes.

# ANEXOS

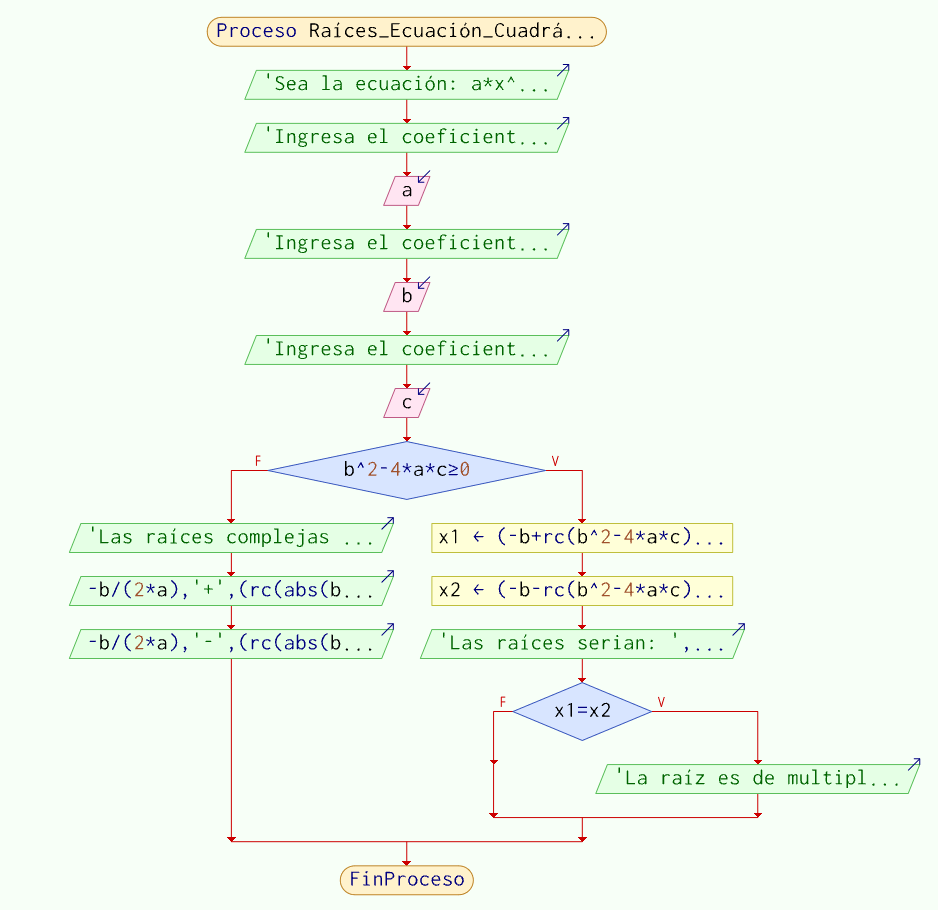
**Diagrama de flujo ejercicio 1.**

****

**Diagrama de flujo ejercicio 2.**



**Diagrama de flujo ejercicio 3.**



**Diagrama de flujo ejercicio 4.**

