



ENES
JURIQUILLA

Ingeniería Aeroespacial

204 - Fundamentos de Programación

Ing. Andrés David Flores Ferriz

Práctica 3

Diagramas de Flujo

Actividad en clase 25 de 25 pts

Protocolo de práctica	56 de 75 pts
Portada	08 de 10 pts
Introducción	10 de 15 pts
Desarrollo	15 de 25 pts
Conclusiones	15 de 15 pts
Referencias	08 de 10 pts

CALIFICACIÓN TOTAL Entrega 81 pts

Equipo 1 Grupo: 2002

Paul Sebastian Gutierrez de Loza
Emilio Lora Aguilar
Marianangel Ramos Garzúlez

Nro. Cuenta

425037061

425117677

425062412

Fecha: 21- febrero - 2025

- Introducción con muy poco contenido, el proposito es rescatar información relevante de clase e investigar algo adicional con detalle
- No se atendieron correcciones en desarrollo y no se rescata para que funciona cada cosa, se describe lo que tiene contenido solamente.
- Como se menciona en clase la idea del protocolo era que conectaran mas toda la técnica y bueno aquí ya solo se encuentra el DFD y solo mencionan las modificaciones que hacen de su tabla pero no la rescatan o profundizan mas en actualizar entradas salidas y modificaciones en el proceso.
- Agregar dato de tipo de archivo consultado en referencias [Web,PDF, Libro, etc]

Introducción:

Los diagramas de flujo son una herramienta muy útil en la resolución de problemas ya que permiten representar visualmente el flujo de ejecución de un algoritmo mediante símbolos gráficos. Su correcto diseño facilita la comprensión e implementación de algoritmos en un lenguaje de programación.

Explorando el funcionamiento de los diagramas de flujo, descubrimos que estos no solo se usan en programación, si no que también se pueden aplicar a otras disciplinas, esto gracias a que su estructura permite visualizar de manera clara procesos y analizar sistemas complejos.

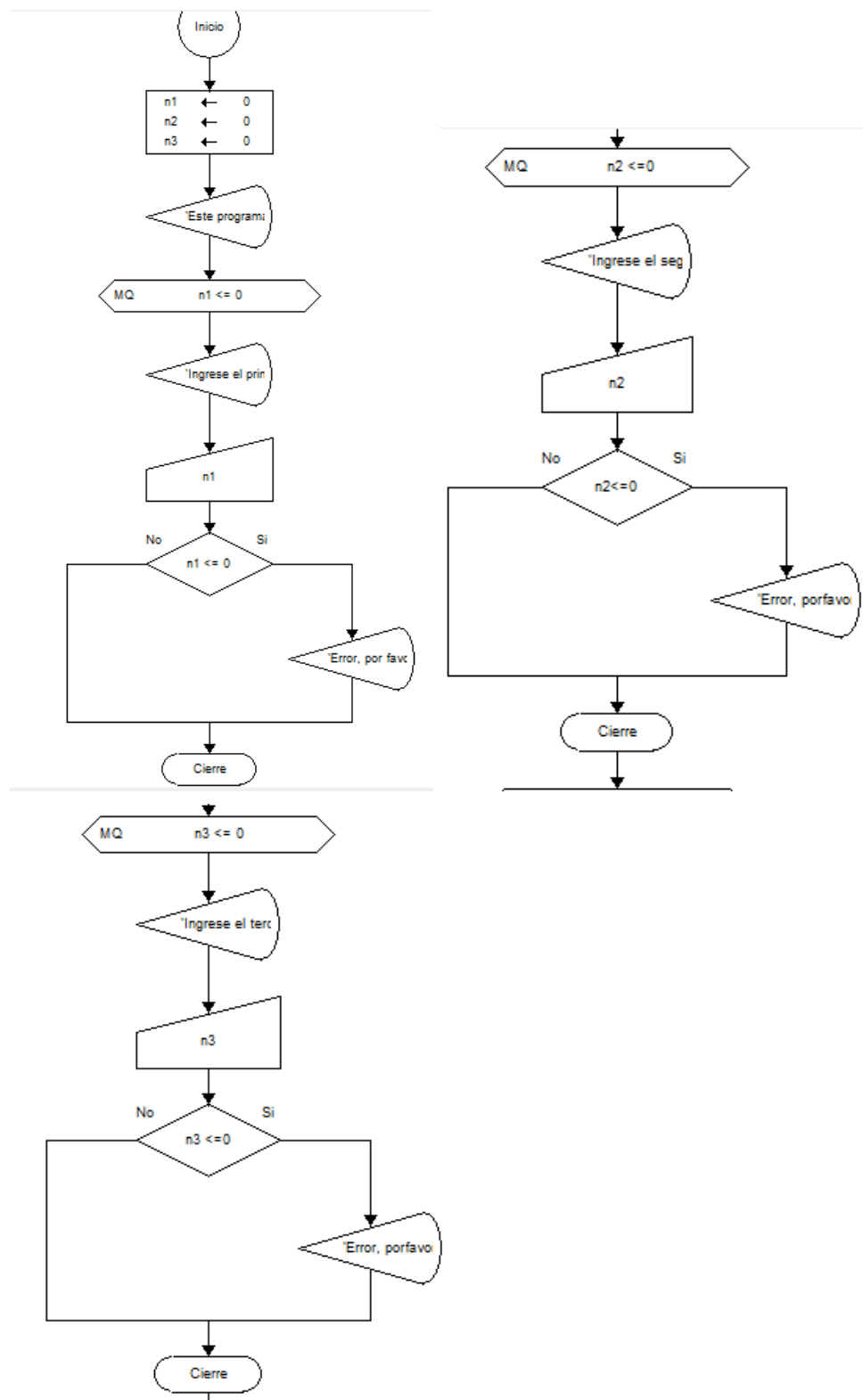
Algunos ejemplos de ramas externas a la programación donde se aplican estos diagramas son la biología, psicología o economía:

Por ejemplo, en la biología se suelen utilizar para representar reacciones químicas en el interior del cuerpo, como la digestión, la respiración, o algunas más complejas como el ciclo de vida de bacterias.

El que los diagramas de flujo se utilicen en otras disciplinas nos muestra la potencia y utilidad que tienen para simular procesos o replicar algoritmos. Con esto entendimos el porqué se aplican en la programación.

Desarrollo:

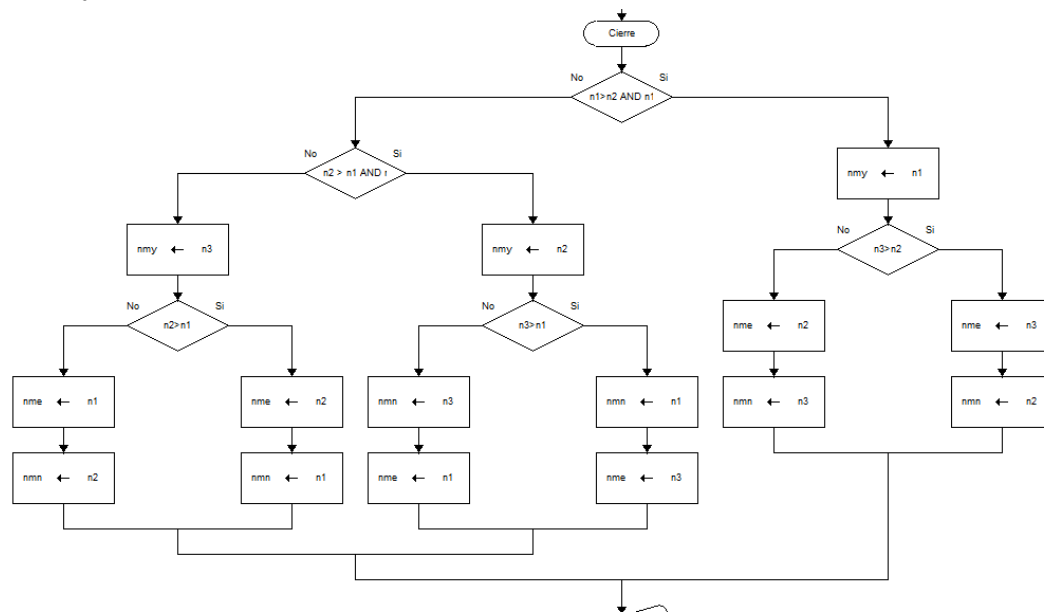
Programa para ordenar 3 números de menor a mayor



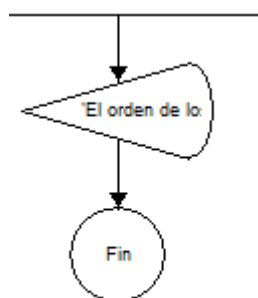
Decidimos que solo fueran 3 números reales sin contemplar a los negativos para agilizar el proceso a la hora de programar. Se mete el primer dato y si es mayor que cero pasa a que se ponga el siguiente número y así con los 3. En caso de que un número sea negativo o 0, este lanzará un anuncio de error y no avanzará hasta que se ponga un valor mayor a 0. Se determinó que también podrían usarse decimales.

En este programa se pusieron más salidas de las que teníamos planeadas por cada entrada de dato que se tenía se creó una salida en caso de que un número puesto no fuese el

correcto. No se tenía contemplado pero llegamos al acuerdo de que así sería mucho más eficiente y más fácil de utilizar.

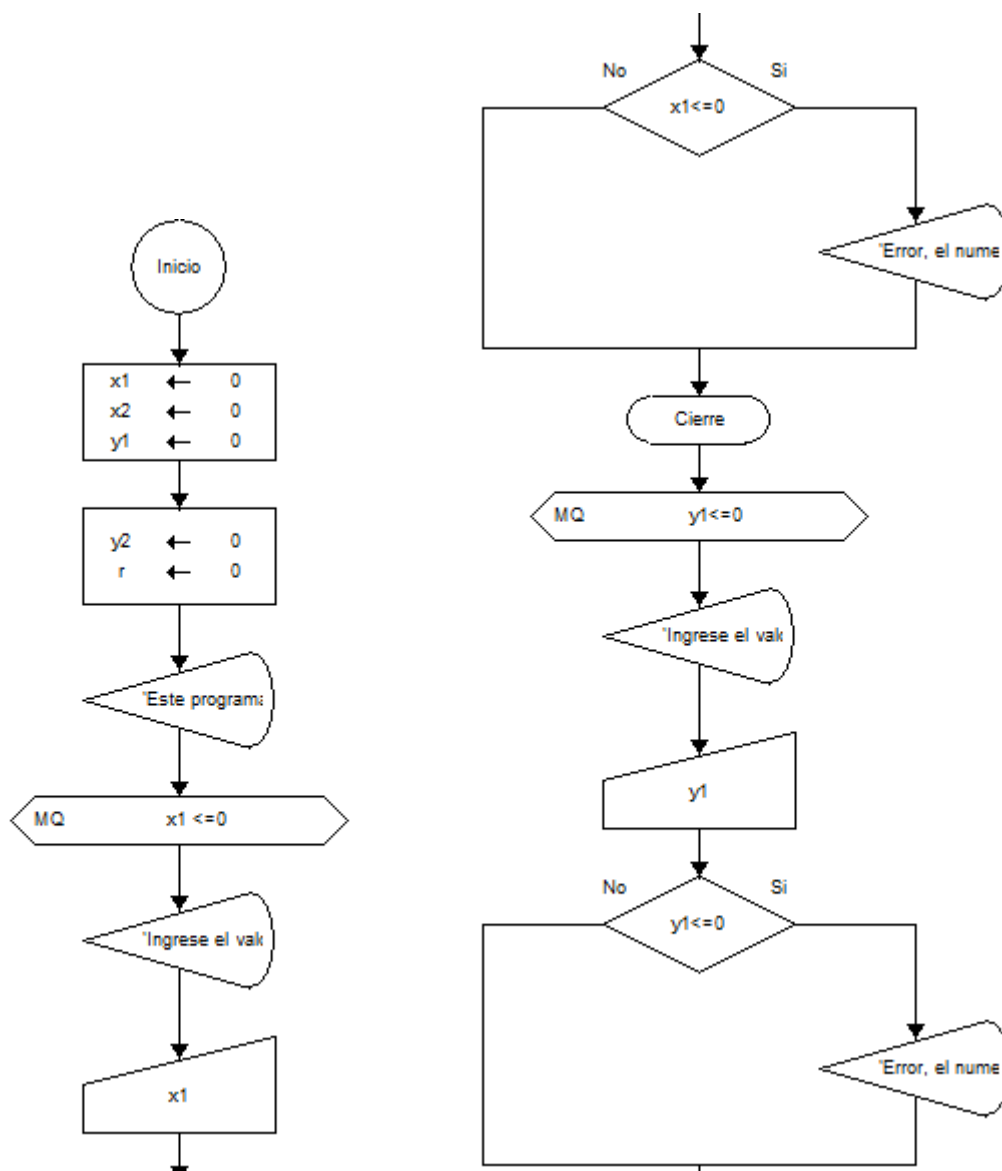


Aquí se analiza, la primera condición es que $n1$ sea mayor que $n2$ y que $n3$, si la cumple pasará a ver si $n3$ es mayor que $n2$, aquí ya se sabe cual es menor y cual es mayor por lo tanto el programa lo organiza y arroja el resultado. En caso de que la primera condición sea falsa llegará a la condición de si $n2$ es mayor que $n1$ y $n2$ mayor $n1$ que $n3$, si la cumple sólo faltaría ver qué número es mayor si $n3$ o $n1$ y así arroja el resultado, en caso de no cumplir con la condición debe de checar cuál es mayor si $n2$ o $n1$ y de ahí ya se arroja el resultado.



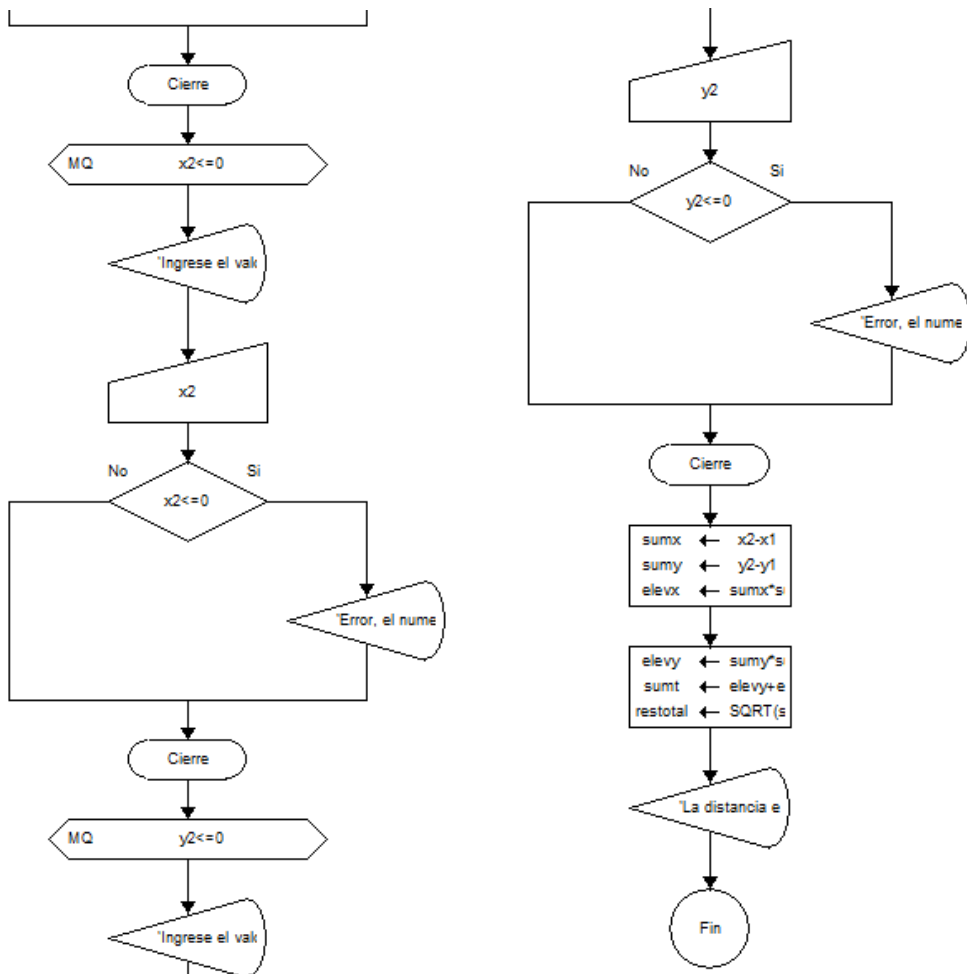
Finaliza con el resultado de esos números ya ordenados de menor a mayor y un mensaje diciendo “El orden de los números es el siguiente: nmn , nme , nmy . Fin del programa”

Programa para calcular la distancia entre dos puntos



En este programa hay 10 variables, 4 para las coordenadas y 6 para operaciones y resultados, va a pedir una por una mientras que cumplan con la condición de mayor a 0, en caso de que no cumpla con esta condición marcará error y no continuará hasta que el número coincida con la indicación.

Se pusieron más salidas de lo que se tenía planeado para que el programa funcionará de manera correcta y también se agregó para que se usarán decimales en caso de que la coordenada no sea un número cerrado.



Toma las variables y las sustituye en la fórmula para calcular la distancia entre dos puntos, raíz de $[(x2-x1)^2 + (y2-y1)^2]$.

Al finalizar te arroja el resultado con la leyenda: “La distancia entre los dos puntos es; restotal, Fin del programa”.

Conclusión:

Paul: El usar diagramas de flujo siempre se me ha hecho una manera más sencilla de empezar a programar y de resolver problemas, ya que permite ver errores, organizar el programa más eficientemente y buscar soluciones rápidas. Disfrute hacer los diagramas de flujo para los problemas en DFD, ya que el programa es muy intuitivo, amigable con el usuario y al mismo tiempo funcional.

Lara: Está práctica fue de mucha utilidad, hacer diagramas de flujo nos permite representar los pasos que debe de seguir el código de manera clara, y DFD es una herramienta muy buena para esto, ya que te permite “ejecutar” el diagrama de flujo y ver si te equivocaste en alguno de los pasos de el algoritmo.

Gel: Fue entretenido porque muchas veces falló solo por una coma y así vimos en qué se estaba fallando para poder arreglarlo rápido y que quedara bien el programa. También vimos que muchas restricciones son difíciles de poner y otras bastante fáciles.

Referencias:

- Burnette, K. A. S. (2020). Drawing Flowcharts of Lab Protocols Helps Students Prepare for Biology Labs. *CourseSource*, 7. <https://doi.org/10.24918/cs.2020.2>
- Kimber, O., Cromley, J. G., & Molnar-Kimber, K. L. (2018). Let Your Ideas Flow: Using Flowcharts to Convey Methods and Implications of the Results in Laboratory Exercises, Articles, Posters, and Slide Presentations. *Journal Of Microbiology And Biology Education*, 19(1). <https://doi.org/10.1128/jmbe.v19i1.1477>