



ENES
JURIQUILLA

UNIVERSIDAD NACIONAL AUTONOMA DE MÉXICO

CAMPUS JURIQUILLA



INGENIERIA AEROESPACIAL.

Práctica 2.

Solución de problemas y algoritmos.

Óscar Abraham Murillo Rivas - 425039773

Sergio Reséndiz Cabrera - 425061336

Saul Emiliano Revoreda Gómez - 425007921

Fundamentos de programación.

Andres David Flores Ferro

Semestre 2

Grupo 2002

Fecha de entrega: 14/02/2025

Fecha de revisión: 02/03/2025

Observaciones y calificación:

- Introducción con muy poco contenido de clase y se agrega algo demasiado breve de investigación adicional a lo mostrado en clase.
- Agregar dato de tipo de archivo consultado en referencias [Web,PDF, Libro, etc]
- En el desarrollo solo se menciona el contenido pero no hay desarrollo de teoría de lo que se hizo o porque (en su caso algunos casos si pero no), especificación indica descripción paso a paso de lo realizado y de su porque.

Actividad en clase 23 de 25 pts

Protocolo de práctica	59 de 75 pts
Portada	10 de 10 pts
Introducción	07 de 15 pts
Desarrollo	19 de 25 pts
Conclusiones	15 de 15 pts
Referencias	08 de 10 pts

CALIFICACIÓN TOTAL Entrega 82 pts

Solución de problemas y algoritmos

Introducción.

En clase se nos indico el como realizar una “tablero”, el cual nos ayuda a detallar un problema asignado, desde nuestro punto de vista es de mucha ayuda ya que antes de comenzar a resolver el problema, nos permite visualizar de manera detallada punto por punto cada cosa a desarrollar, que salidas y entradas debe de haber e incluso las restricciones que puede haber en el programa. Es decir, hacer el tablero es un paso fundamental para poder construir un algoritmo con ciertos requerimientos disminuyendo su complejidad.

Tras investigar en algunas páginas, todas coinciden con lo mismo, lo primero con lo que debes de empezar es con la lectura y comprensión del problema, ya que esto es lo que permite tener una visión de a donde se quiere llegar con el programa y si tiene o no restricciones. Es importante recalcar que antes de escribir una sola línea de código se debe hacer todo este procedimiento de manera detenida.

Nosotros como equipo añadiríamos que dentro de los pasos a seguir se pueden incluir algunos tips o trucos para disminuir aun más la complejidad y hacer el procedimiento menos abstracto, es decir, en pocas palabras se trata de hacer un desglose más detenido del problema.

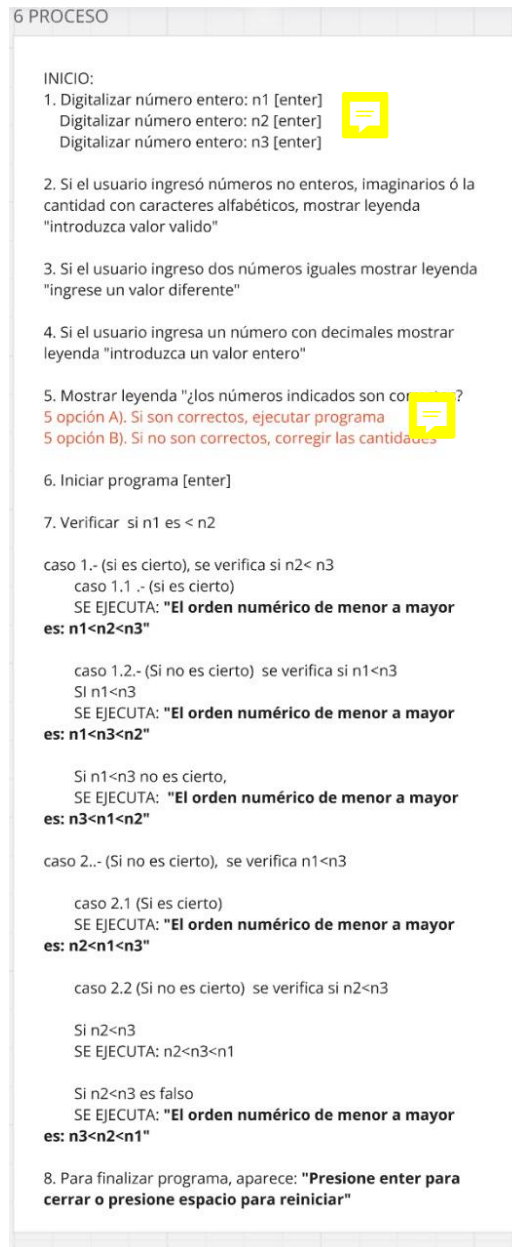
Desarrollo (Evidencias de práctica y desarrollo).

- Problema 1 (Ordenar 3 números de menor a mayor)

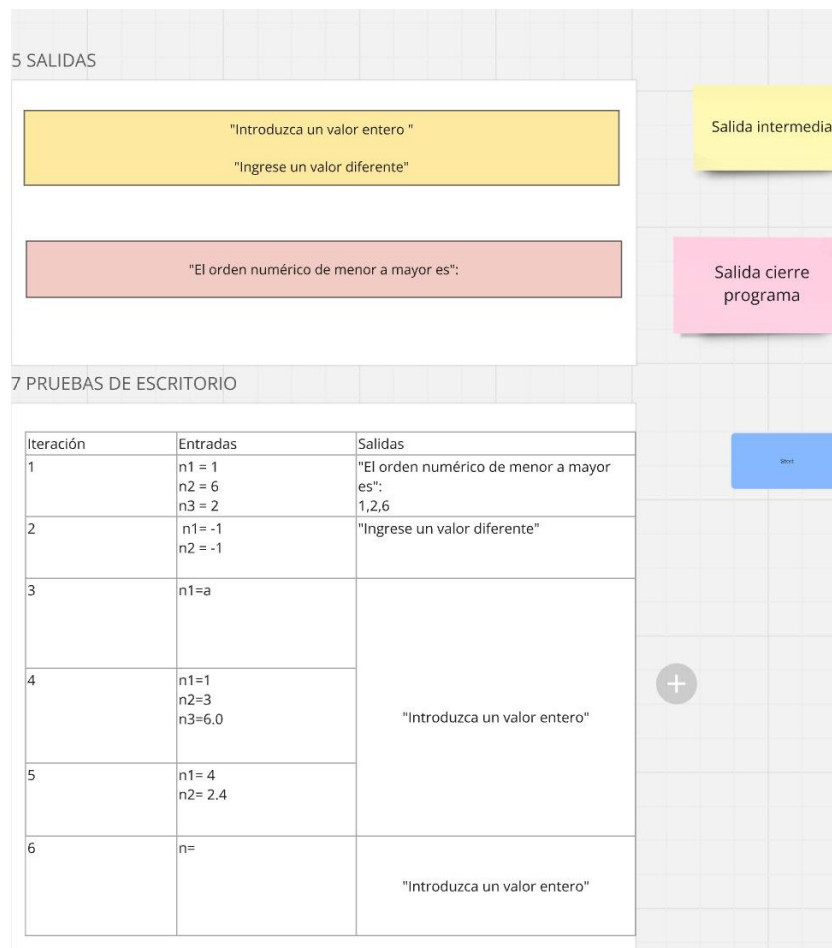
Aquí adjuntamos la foto de la primera parte de la práctica, en donde se pueden ver los primeros 4 pasos del tablero hecho para resolver el problema de ordenar 3 números de menor a mayor. En el paso uno planteamos el problema. En el paso dos ponemos los requerimientos para que el algoritmo funcione de manera correcta en dónde también ponemos algunas propuestas para darle un objetivo más claro en cuanto a la resolución del problema. En el paso tres **precedimos** a poner ciertas restricciones que van a limitar el algoritmo para evitar obtener resultados imprecisos y por ultimo en el paso cuatro adjuntamos las entradas posibles para el usuario.



En la siguiente imagen podemos ver el procedimiento seguido (algoritmo) para poder resolver el problema, cabe recalcar que es necesario poner todas las operaciones posibles que puede tener el programa guiándonos en nuestros requerimientos y restricciones, para tener en orden un procedimiento claro que no genere confusiones al momento de intentar darle solución al problema.

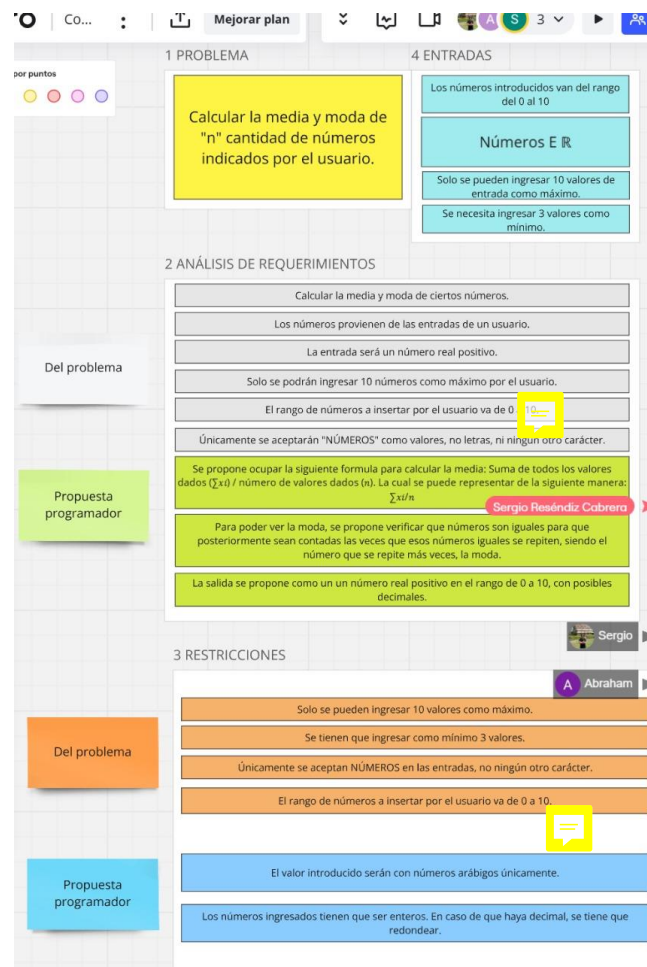


En la siguiente imagen podemos ver las salidas posibles para el usuario, estas igualmente están hechas guiadas en los requerimientos y las restricciones que tiene el problema, se puede ver que habrá salidas intermedias en el programa y una salida final la cual será el resultado de la resolución del problema. Posteriormente podemos ver que hay un apartado dedicado únicamente a las pruebas de escritorio, estas pruebas tienen que hacerse evaluando las posibles entradas del usuario, dando como resultado las salidas correspondientes. Es en este último paso que se evalúa la funcionalidad del algoritmo.



- Problema 2 (Calcular la media y moda de “n” cantidad de números)

Aquí podemos ver la imagen de los primeros pasos a seguir para resolver el problema de calcular la media y moda de “n” cantidad de números indicados por el usuario. En el paso uno se plantea el problema. En el paso dos hacemos un análisis de requerimientos para que el algoritmo funcione de manera correcta, es aquí donde también damos una propuesta de cómo podemos solucionar el problema para hacerlo de la manera más simple posible. En el paso tres definimos algunas restricciones que va a tener el algoritmo, para así poder dar una resolución concisa y acertada del problema, que no tenga variaciones. En el paso cuatro ponemos únicamente las entradas posibles para el usuario.




En la siguiente imagen podemos ver como tal los procedimientos a seguir para poder darle resolución al problema dado. Es indispensable poner TODOS los pasos de manera ordenada y concisa para que no haya confusiones al momento de hacer el procedimiento para encontrar la solución.

6 PROCESO

INICIO:

1. Iniciar programa y mostrar leyenda "Ingrese los valores en cada una de las celdas correspondientes".
2. Iniciar programa, y mostrar las casillas en donde se pondrán los valores.
n1:
n2:
n3:
n4:
n5:
n6:
n7:
n8:
n9:
n10:



2. Si el usuario ingresó una cantidad no permitida por el programa en cualquier n, mostrar leyenda "Ingresar únicamente valores permitidos".
3. Mostrar leyenda "¿Los valores indicados son correctos?"

3 Opción A) Si son correctos, iniciar programa
3 Opción B) Si no son correctos, reingresar los valores
4. Ejecutar programa [enter]

Sergio

5. El programa procede a desplegar dos leyendas en dónde aparezcan las dos salidas: media y moda.
6. En caso de que si exista media y moda, aparecerá el resultado en las leyendas correspondientes que dirán:

Texto 1: "La moda de los valores indicados es:"
Texto 2: "La mediana de los valores indicados es:"
7. Si todos los números tienen la misma frecuencia de aparición, en el texto correspondiente aparecerá la siguiente leyenda:

Texto 1: "La moda de los valores indicados es: Todos los números tienen la misma frecuencia de aparición, no hay moda".

En la siguiente imagen podemos ver las salidas posibles para el usuario dependiendo de la entrada que haya introducido y las cuales están definidas por los requerimientos y las restricciones que tenemos del problema, se puede notar a simple vista que habrá salidas intermedias en el algoritmo, así como también se puede ver que habrá una salida final, en la cual se nos mostrará la solución del problema. Posteriormente se pueden ver las pruebas de escritorio, en las cuales se pueden poner las posibles entradas de usuario ya sean correctas o incorrectas para así dar como resultado las salidas correspondientes, probando la correcta o incorrecta funcionalidad del algoritmo.

Copia de 5 SALIDAS

"Indicar únicamente valores permitidos"
"¿Los valores indicados son correctos?" Opción A) Si son correctos, iniciar programa Opción B) Si no son correctos, reingresar los valores
"Todos los números tienen la misma frecuencia de aparición, no hay moda".
El valor introducido es incorrecto, verificar y volver a ingresar "
"La moda de los valores indicados es": "La mediana de los valores indicados es": "La moda de los valores indicados es: Todos los números tienen la misma frecuencia de aparición, no hay moda".

Sergio

Pruebas de escritorio

Iteración	Entradas	Salidas
1	n1= 2 n2=3 n3=5 n4=8 n5= 2 n6= 5 n7 = 7 n8= 10 n9= 0 n10= 2	media=4.4 moda= 3
2	n1=a	"El valor introducido es incorrecto, verificar y volver a ingresar."
3	n1 = 0	A Abraham
4	n1=1 n2=3 n3= 5.5	"El valor introducido es incorrecto, verificar y volver a ingresar."
5	n1=3 n2= 11	

Sergio Reséndiz Cabrera

Conclusiones.

Sergio: Dentro de la práctica del día viernes 7 de Febrero, vimos los pasos a seguir para poderle dar solución dos problemas determinados poniendo en practica los conociminetos previamente adquiridos. Esta práctica nos sirvio para poder hacer un analisis detallado para poder resolver un problema determinando desglosando todos sus pasos a seguir haciendo un algoritmo, es importante saber hacer esto debido a que es uno de los pasos que preceden el hacer un codigo de programación y sirve para no generar confusiones y llevar un orden en el proceso de generar el código.

Abraham: En esta práctica aprendí cómo elaborar de manera ordenada algoritmos y cómo distribuirla en etapas para la resolución de problemas lógicos de software, posteriormente implementaremos este nuevo eprendizaje en el proceso de génerar codigos con un lenguaje de programación (Lenguaje C), y asi como estudiantes de ingenieria, sabiendo utilizar estas herramientas podremos darle solución a un problema determinado.

Saul: En esta práctica me estrese un poco porque aun no se bien como programar un algoritmo, por lo que no sabíamos como realizar los códigos sin embargo, uno de mis compañeros que tiene más conocimiento del tema fue el que pudo realizar ese procedimiento sin muchas complicaciones. Me gusto la forma en la que se organiza la tabla ya que ofrece una mayor organización en cuanto a la solución del problema descrito.

Referencias.

Casero, A. (2024, 15 marzo). Principios de resolución de problemas en programación.

KeepCoding Bootcamps. <https://keepcoding.io/blog/resolucion-de-problemas-en-programacion/>

Squirrels, J. (2024, 29 mayo). *Resolución de problemas. Cómo aumentar su capacidad para resolver tareas y desafíos de programación.* CodeGym.

<https://codegym.cc/es/groups/posts/es.523.resolucion-de-problemas-como-aumentar-su-capacidad-para-resolver-tareas-y-desafios-de-progra>