



**ENES
JURIQUILLA**

**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
ESCUELA NACIONAL DE ESTUDIOS SUPERIORES UNIDAD JURIQUILLA (ENES
JURIQUILLA)**

Ing. Aeroespacial

Segundo semestre

Santiago de Querétaro, Qro. 11 de febrero del 2025

Trabajo:

Práctica 2

“Solución de problemas y algoritmos”

Materia:

Fundamentos de Programación

Presenta:

Serrano Araujo Matías, No. de cuenta: 425030550

Reyes Vargas Edgar, No de cuenta: 322335080

Docente:

Ing. Andres David Flores Ferro

Fecha de revisión:

Observaciones:

Calificación:

Introducción:

Durante esta práctica nuestro equipo realizará un algoritmo en la plataforma “**Miro**” de manera correcta y siendo eficaz para poder dar solución a los problemas dados por el profesor en clase, en el que seguimos las etapas de Análisis y Diseño pertenecientes al Ciclo de vida del software.

Un algoritmo es una secuencia de instrucciones u operaciones específicas que permiten controlar determinados procesos. Se trata de conjuntos finitos y ordenados de pasos, que nos conducen a resolver un problema o tomar una decisión, para realizar un algoritmo para solución de problemas en Miro, tenemos que cumplir con ciertas características:

Preciso: Debe indicar el orden de realización de paso y no puede tener ambigüedad.

- Definido: Si se sigue dos veces o más se obtiene el mismo resultado.
- Finito: Tiene fin, es decir tiene un número determinado de pasos.
- Correcto: Cumplir con el objetivo.
- Debe tener al menos una salida y ésta debe de ser perceptible
- Debe ser sencillo y legible
- Eficiente: Realizarlo en el menor tiempo posible.

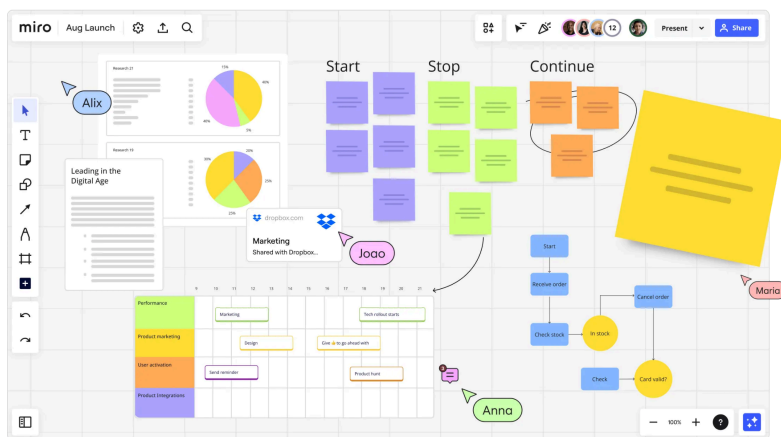
Para poder hacer un algoritmo, primero se tiene que definir un problema informático en este caso que queramos resolver, y este es un conjunto de peticiones o instancias a las cuales se les da o les corresponde un conjunto de soluciones. Para que resolvamos los problemas nos apoyamos en la ingeniería de software, en el que esta se define como *“una secuencia de instrucciones u operaciones específicas que permiten controlar determinados procesos. Se trata de conjuntos finitos y ordenados de pasos, que nos conducen a resolver un problema o tomar una decisión”* que esto nos sirve para tener un software fiable y eficiente.

Ahora, para la solución de problemas, requerimos de un ciclo de vida de software, en el cual su contenido es una referencia de las tareas realizadas para el desarrollo de un software, este ciclo de vida contiene: definición de necesidades, análisis, diseño, codificación, pruebas, , validación y por ultimo mantenimiento y evolución, y durante este ciclo de vida, es de suma importancia entender las necesidades requeridas y definirlas de manera clara, para eso identificamos los dos conjuntos dentro del sistema, el de entrada y el de salida donde el conjunto de entrada está compuesto por todos aquellos datos que pueden alimentar al sistema y el conjunto de salida está compuesto por todos los datos que el sistema regresará como

resultado del proceso. Estos datos se obtienen a partir de los datos de entrada. Esto lo aplicamos a la hora de realizar nuestros algoritmos para la solución de problemas matemáticos en el que definimos cada variable, restricciones y conjuntos de entrada y salida en los siete pasos correspondientes. Estos algoritmos nos servirán para posteriormente plasmar esta información en un diagrama de flujo, el cual es un diagrama que describe un proceso, sistema o algoritmo informático. Se usan ampliamente en numerosos campos para documentar, estudiar, planificar, mejorar y comunicar procesos que suelen ser complejos en diagramas claros y fáciles de comprender.

Desarrollo:

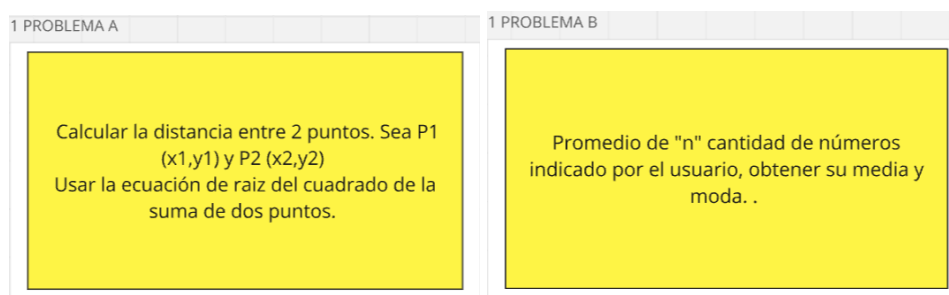
El primer paso en el desarrollo de resolver estas actividades fue el elegir una plataforma en la que poder trabajar y aportar a la meta de crear el algoritmo de forma conjunta, debido a sugerencia del profesor se decidió utilizar Miro, la cual es una plataforma online de colaboración visual, también conocida como pizarra online, la cual le permitió al equipo realizar modificaciones de forma colaborativa a los distintos algoritmos.



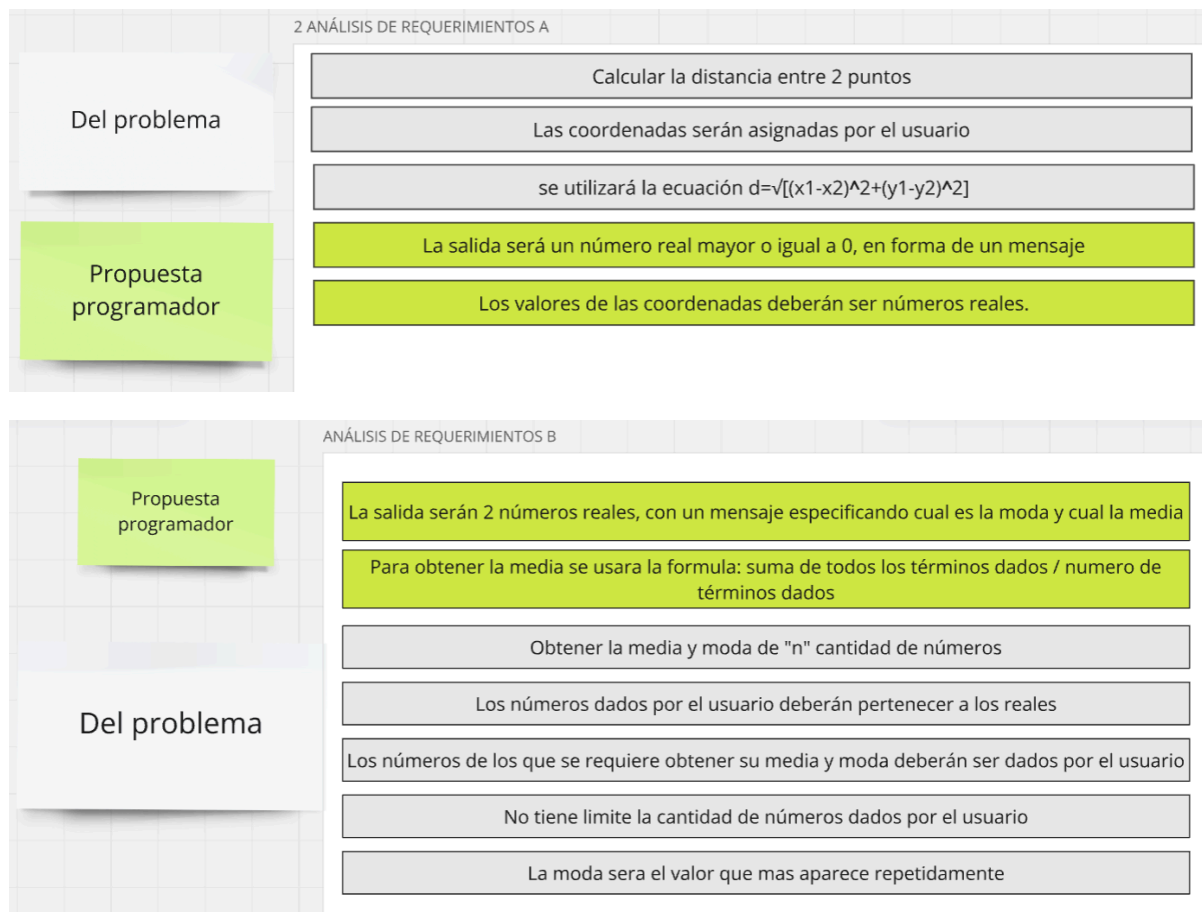
La actividad a realizar esta práctica fue elaborar dos algoritmos que resuelvan dos problemas determinados por el profesor, identificando los módulos de entrada, de procesamiento y de salida.

Para poder conseguir elaborar estos algoritmos debemos seguir el ciclo de vida del software empezando con el primer paso, la definición de necesidades, dentro de este lo primero que necesitamos hacer fue definir los problemas a resolver.

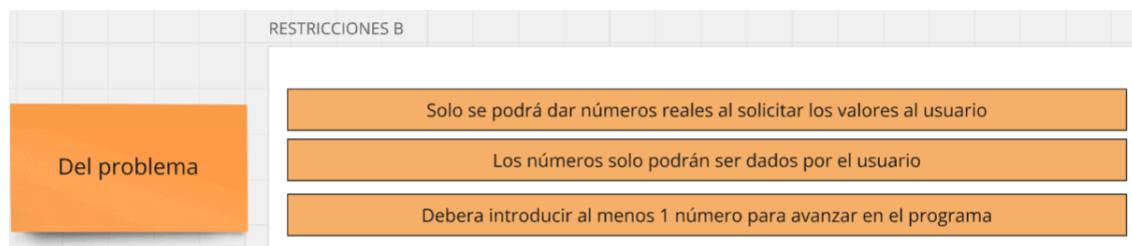
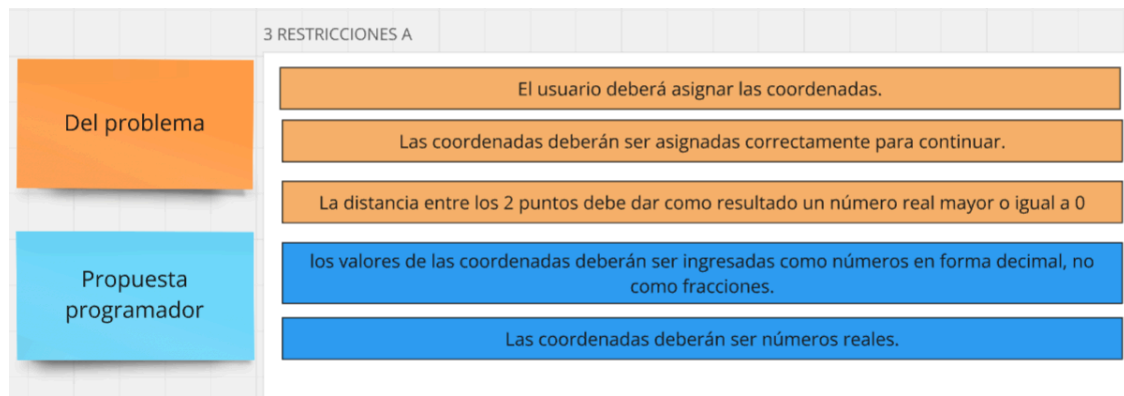
- A. Calcular la distancia entre 2 puntos. Sea $P1 (x1,y1)$ y $P2 (x2,y2)$, utilizando la ecuación de raíz del cuadrado de la suma de dos puntos.
- B. Obtener la media y moda de "n" cantidad de números indicado por el usuario.



Posteriormente, continuando con la definición de necesidades debemos identificar los requerimientos establecidos por el problema o el programador, los cuales serán las metas a cumplir para poder considerar el funcionamiento del algoritmo como exitoso, incluirá las condiciones establecidas por el problema y condiciones secundarias para poder cumplirlas.

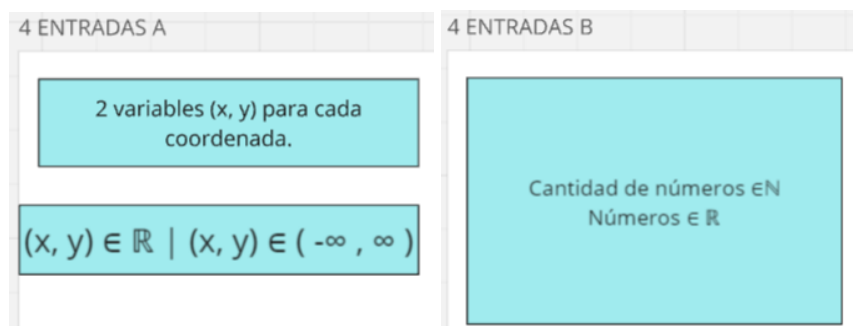


Para terminar la definición de necesidades y comenzar con el análisis debemos darnos cuenta de las restricciones, condiciones específicas a cumplir para permitir la realización correcta de los requerimientos y el buen funcionamiento del algoritmo.

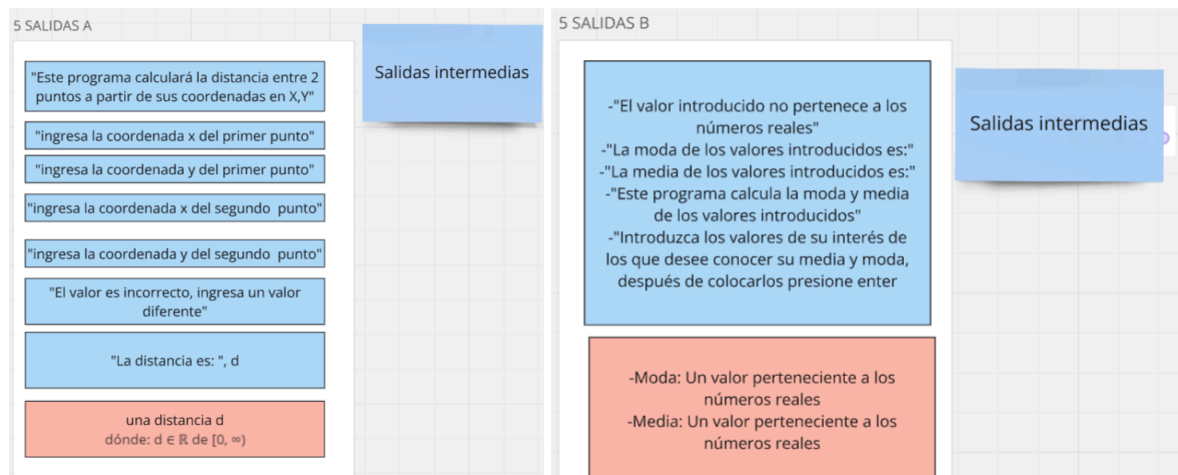


La segunda etapa del ciclo de vida del software es el análisis, el cual busca entender la necesidad del problema, entenderlo, busca entender qué es lo que requiere el usuario, busca un entendimiento más profundo de los requerimientos iniciales con el objetivo de no caer en ambigüedades, para esto se requiere identificar 2 grandes conjuntos, el de entrada y el de salida.

Comenzaremos identificando el conjunto de entrada que incluye todos aquellos datos que pueden alimentar al sistema, tanto las constantes como los que son determinados por el usuario, debemos identificar qué tipo de datos, o de qué conjunto de datos estos son miembros. Por ejemplo en el problema A las variables “x” y “y” pertenecen a cualquier número real, al representar coordenadas en un plano cartesiano, las cuales pueden ser decimales. Mientras que en el problema B la cantidad de números que el usuario ingresará deben pertenecer a los naturales (enteros y positivos), al representar la cantidad de datos.



El siguiente conjunto es el de salida, los cuales son los datos obtenidos mediante el proceso o algoritmo a partir de el conjunto de entradas, como el resultado de los cálculos, estas también incluirán toda la información proporcionada al usuario de parte del algoritmo por ejemplo mensajes, mensajes de estado, retroalimentación o respuesta a ciertas acciones. Este puede o no ser visible para el usuario y pueden manifestarse como; texto, tablas, de forma gráfica, numérica, acciones o acciones externas como enviar datos a otras aplicaciones.



El tercer paso del ciclo de vida del software es el diseño, en el cual diseñaremos el algoritmo que será utilizado para cumplir los requisitos y resolver el problema, este es el proceso o la solución que transforma los datos de entrada en las salidas que necesitamos.

Un algoritmo es la parte más importante de las ciencias de la computación y para asegurar que sea un buen algoritmo debe ser correcto (cumplir con el objetivo) y eficiente (realizarlo en el menor tiempo posible), además de ser entendible para cualquier persona.

Para conseguir esto debemos proponer alternativas viables para la solución del problema, teniendo en cuenta los resultados del análisis para así construir una serie de pasos que llevarán al resultado deseado, utilizando distintas herramientas como las variables que almacenan los distintos valores y datos necesarios para el proceso.

6 PROCESO A

nota: avanzar al presionar enter

1. Declarar 5 variables. La distancia d, y las coordenadas: x1, y1, x2, y2
2. mostrar: "Este programa calculará la distancia entre 2 puntos a partir de sus coordenadas en X,Y"
3. mostrar: "ingresa la coordenada x del primer punto"
4. leer y verificar el valor ingresado, en caso de no cumplir las restricciones señaladas ir al paso 17
5. Asignarle a la variable x1 el valor ingresado por el usuario
6. mostrar: "ingresa la coordenada y del primer punto"
7. leer y verificar el valor ingresado, en caso de no cumplir las restricciones señaladas ir al paso 18
8. Asignarle a la variable y1 el valor ingresado por el usuario
9. mostrar: "ingresa la coordenada x del segundo punto"
10. leer y verificar el valor ingresado, en caso de no cumplir las restricciones señaladas ir al paso 19
11. Asignarle a la variable x2 el valor ingresado por el usuario
12. mostrar: "ingresa la coordenada y del segundo punto"
13. leer y verificar el valor ingresado, en caso de no cumplir las restricciones señaladas ir al paso 20
14. Asignarle a la variable y2 el valor ingresado por el usuario
15. asignarle a la variable d el resultado de la operación: $\sqrt{(x1-x2)^2+(y1-y2)^2}$
16. mostrar: "La distancia es: ", d
- Fin del programa**
17. mostrar: "El valor es incorrecto, ingresa un valor diferente", ir al paso 3
18. mostrar: "El valor es incorrecto, ingresa un valor diferente", ir al paso 6
19. mostrar: "El valor es incorrecto, ingresa un valor diferente", ir al paso 9
20. mostrar: "El valor es incorrecto, ingresa un valor diferente", ir al paso 12

6 PROCESO B

Inicio:

- 1.-Se muestra el mensaje del inicio de programa: "Este programa calcula la moda y media de los valores introducidos"
-El programa no avanzara a menos que el usuario introduzca un número real
- 2.-"cuál es la cantidad de números a introducir"
- 3.-leer y guardar el valor
- 4.-Aparecerá el siguiente mensaje: "Introduzca los valores de su interés de los que desee conocer su media y moda, después de colocarlos presione enter".
- 5.-Se guardará el valor introducido por el usuario, se repetirá el número de veces introducido por el usuario.
- 6.-En caso de que los valores introducidos no cumplan con las restricciones aparecerá el siguiente mensaje: "El valor introducido no pertenece a los números reales"
-Se regresara al inicio del programa para volver a introducir los valores
- 7.-La media se obtendrá como: La sumatoria de todos los términos / el número de términos.
- 8.-La moda será el valor que más aparece repetidamente
- 9.-Aparecerán los siguientes mensajes:
-"La moda de los valores introducidos es:" moda
-"La media de los valores introducidos es:" media

FIN DEL PROGRAMA

Por último realizamos la valoración del algoritmo mediante pruebas de escritorio, utilizando datos representativos del problema siguiendo el algoritmo para verificar su funcionamiento, realizandolo varias veces o iteraciones para poder confirmarlo con una mayor certeza.

7 PRUEBAS DE ESCRITORIO A		
Iteración	Entradas	Salidas
1	x1, y1: 1,1 x2, y2: 2,2	1.4142
2	x1, y1: 1,1 x2, y2: 1,1	0
3	x1, y1: 5,7 x2, y2: 4,3	4.1231
4	x1: at	hace que el usuario reasigne un valor hasta detectar un valor adecuado.
5	x1, y1: 1,0 x2, y2: 9,0	8

7 PRUEBAS DE ESCRITORIO B		
Iteracion	Entrada	Salida
1	1, 3, 3, 5, 3, 7	La moda de los valores introducidos es: 3 La media de los valores introducidos es: 3.67
2	1.5, 6, 8, 8, 8, 9, 13, 4	La moda de los valores introducidos es: 8 La media de los valores introducidos es: 7.19
3	1, 2, a, #, 7	El valor introducido no pertenece a los números reales
4	5	La moda de los valores introducidos es: 5 La media de los valores introducidos es: 5
5	4, 6, ?, f, %	El valor introducido no pertenece a los números reales

Conclusiones:

Matías- considero que la práctica fue muy interesante, y en mi caso personal bastante útil, al poder darme cuenta de todos los pasos que conlleva la realización de un algoritmo, lo que se puede trasladar a la resolución de cualquier problema fuera de la programación. Y si bien son cosas que quizá pude haber realizado de forma no tan consciente creo que al analizar y estructurar los pasos individualmente puedes conseguir un entendimiento bastante más profundo de estos.

Edgar: Para alguien que tiene muy poca experiencia en este tema de programación como yo, me parece que estas prácticas para ir adquiriendo esa lógica de programación me parecen muy buenas ya que te van diciendo paso a paso todas las variables que debes tomar en cuenta a la hora de querer resolver un problema por más sencillo que parezca que eso es lo que en mi caso más se me dificultó ya que tengo nula experiencia en programación la verdad es que todavía no puedo definir todas las variables existentes o restricciones para un problema pero considero que es algo que se gana con la experiencia, además que es lo que debe llevar un buen software y te dan las bases para saber qué es lo que debe llevar un buen software y empezar a codificar.

Referencias:

- E García Cano, E., & A Solano Gálvez, J. (2025, 22 enero). Guía práctica de estudio 03: Solución de problemas y Algoritmos. Manual de Prácticas del Laboratorio de Fundamentos de Programación UNAM.
- Arbolea, J. (2023, 2 mayo). Informática = Entrada-Proceso-Salida. Velneo.
<https://www.velneo.com/blog/informatica-entrada-proceso-salida>
- Ramirez, J. (2024, 28 octubre). Formatos de salida: Entiende cómo funcionan los campos y anchos de campo en la entrada y salida de datos. Saberpunto.
<https://saberpunto.com/tecnologia/formatos-de-salida-entienda-como-funcionan-los-campos-y-anchos-de-campo-en-la-entrada-y-salida-de-datos>
- Vizcaíno Alcantud, P. (s. f.). ¿QUÉ ES MIRO? [Diapositivas; Digital]. Universidad de Alicante.
<https://economicas.ua.es/es/documentos/miro.-pizarra-colaborativa.pdf>
- Equipo editorial, Etecé. (2024, 25 noviembre). *Algoritmo en Informática - Concepto, partes y ejemplos*. Concepto.
<https://concepto.de/algoritmo-en-informatica/>
- *Qué es un diagrama de flujo*. (s. f.). Lucidchart.
<https://www.lucidchart.com/pages/es/que-es-un-diagrama-de-flujo>