

# Universidad Nacional Autónoma de México



## Facultad de Ingeniería

Asignatura: Estructura de datos y  
algoritmos I.

Actividad 2: Lo que aprendí en  
fundamentos de programación.

Alumno: Carrasco Ruiz Alan Uriel.

Fecha (10/03/2021)

## Lo que aprendí en fundamentos de programación

Yo en la materia de fundamentos de programación, vimos varios temas, pese a que íbamos más adelantados en teoría que en las prácticas, por lo tanto el laboratorio nos servía para repasar los conocimientos teóricos que ya llevábamos, esto me parecía una buena manera de aprender la práctica, ya que nos llevaba a hacer ejemplos un poco más complicados que los de las prácticas, usualmente siempre hacíamos por lo menos 3 ejemplos parecidos en todo el curso, pero para cada práctica usábamos instrucciones diferentes, o aplicamos los ciclos, por lo tanto cada práctica era aplicar el tema del título de la práctica.

El primer tema que vimos fue el de crear un repositorio para subir nuestros archivos en línea, esa fue una práctica individual, la plataforma que usamos para crearlo fue una llamada GitHub, en la cual tuvimos que crear una cuenta nueva para poder tener acceso a todas las herramientas que te ofrecen, posteriormente se comentó entre el profesor y nosotros algunos buscadores que usamos como Chrome, Mozilla, entre otros, esto para debatir ventajas y desventajas de cada navegador, y la práctica se la mandamos al maestro vía GitHub le compartimos el enlace.

La práctica 2 constó de previamente instalar una máquina virtual en nuestro equipo, para poder conocer la interfaz que usa Linux, cada quién instale el Linux de su preferencia, en mi caso fue Linux mint, esto lo usamos sobre todo para usar el modo consola que nos ofrece, ya que desde ahí podemos escribir y compilar nuestros propios códigos, el ejemplo que vimos básico fue el hola mundo, pero la actividad fue ver para que servía cada comando, por ejemplo para abrir un archivo, para ver las carpetas, para guardar documentos creados desde la consola, etc, e hicimos una carpeta por cada materia, y dentro de cada una materia metimos 5 conceptos básicos de cada una básicamente fue echar un vistazo a Linux.

Práctica 3 consistió en trabajar en Pseint y hacer algunos ejercicios para practicar, e irnos introduciendo a cómo funcionaba la estructura de la programación, al igual

de ver nuestra mejor opción para darle resolución a algún problema que se nos presente, en este caso fueron problemas sencillos, ya que para algunos esto era muy nuevo y nunca lo habían manejado, por otra parte vimos cómo podíamos hacer funcionar nuestros programas escribiendo o seleccionando nuestra sentencia que ocupamos, y solo rellenándola con los datos que usamos era algo básico que nos sirvió para introducción.

Práctica 4 como parte de la documentación de un programa entran aspectos como los diagramas de flujo, los cuales representan nuestro código, pero de una manera gráfica, es totalmente la función del programa, con sus instrucciones de control, pero en forma de diagrama, esto usualmente lo usarías para irte guiando a la hora de escribir tu código, pero en pseint nos daba la opción de generar este tan solo con un botón, la función te ahorra el tener que hacerlo a mano, parte por parte, esta práctica nos sirvió más que nada para llevar un orden y la hora de hacer nuestro diagrama de flujo ser organizados.

Práctica 5 en esta práctica vemos en particular lo que es el pseudocódigo, que muestra en forma de texto los pasos a seguir para solucionar un problema. El pseudocódigo posee una sintaxis propia para poder realizar la representación del algoritmo realizado anteriormente en papel, hicimos varios ejercicios de práctica, que esta me parece fue una de las más fáciles ya que esto lo habíamos hecho anteriormente, y fue como un juego para nosotros.

Práctica 6 esta práctica fue una de las que mas llamo mi atención, ya que al principio tuve miedo porque compilaríamos un programa directamente en la consola de Linux, algo que ni habíamos hecho, pero retomando nuestros comandos básico que vimos al principio, me ayudo a tener confianza, y después de unos días pudimos entregar la práctica, esta práctica nos sirvió para conocer como es compilar un programa en Linux, como debes de escribir como si fuera una Ide, solo que tenías que tener cuidado con apretar algo cuando no estuvieras

en modo escritura, ya que todo tu programa se podía arruinar, también independientemente encontré una solución a eso, que fue que puedes crear un archivo directamente en la carpeta, terminando su nombre con .cpp, posteriormente lo editabas desde aquí y ya no importaba si tenías errores era fácil corregirlos, pero de una u otra manera ya conozco de que maneras se puede compilar un programa en Linux hecho por nosotros mismos.

Práctica 7 esta práctica vimos lo que son las funciones, las cuales nos sirven dependiendo el caso ya sea para no tener que escribir todo el código nuevamente, en el caso de los menús dentro de algún ciclo for, o en el caso de tener que repetir muchas veces las mismas instrucciones, con este método solo llamamos a la función y aparecerán todas las instrucciones que contiene, por otra parte lo podemos ver por tema de estética ya que nuestro método principal se ve mas bonito con muy pocas líneas de código, solo invocando a la función, está la teníamos que declarar fuera del método main e introducir todo lo que iba a requerir ya fuera un dato, y dentro de cada una podíamos usar variables locales en estos casos, o la otra era en el método principal pedir las variables que se solicitan, y con ayuda de la función jalar los valores de esas variables y hacer su función; en este caso a la función le tendríamos que indicar de que tipo eran los valores recibidos, de tipo flotante, carácter o entero; y le daba igual si en la función estaban declaradas como x, y, y en el método principal como a, b; ya que al primer valor lo asociaría con la primer variable.

Práctica 8 aquí vimos lo que son las estructuras de selección, en las cuales nos enfocamos en lo que es el switch, if, if-else e if-else anidados, respecto al switch nos permite determinar acciones dependiendo el caso que sea elegido, para un menú siento que es la mejor opción ya que el usuario que le pediremos al usuario es un número entero fácil de validar, y dependiendo de su selección simplemente se mostraran las instrucciones de ese caso; Es preferible cuando los datos son enteros o conocidos, por ejemplo algún mes, día algún número; Pero por otra parte tenemos

al if que nos servirá para determinar si algo es verdadero o no; el if-else, con el cual sabremos si algo es verdadero o falso, o se cumple o no, dependiendo del camino que se valide se ejecutarán esas sentencias; finalmente el if-else anidados, que es dentro de una condicionante meter otra condicionante, o validar cada variable de nuestro programa; Esto también es útil para hacer menús, pero es más tedioso o es más código, pero para cuando queremos validar varias condiciones es mil veces mejor esto que usar el switch ya que no será tan fácil recurrir a él tal vez en un quiz.

Práctica 9 las estructuras de repetición como lo son los ciclos for, while y do-while, estas nos sirven para que se repitan ciertas series de instrucciones hasta que se cumpla una condición o hasta que cierta variable incremente hasta cierto valor, los ciclos for, me parece que son a los cuales con mayor frecuencia recurrimos a ellos, estos pueden ser declarados sin saber el valor de la variable contadora y ponérselo dentro del ciclo, son super útiles para hacer arreglos o llenar matrices e incluso para llenar estructuras, regularmente estas variables o variables contadoras son las que determinarán cuando se cierre el ciclo; Por otra parte el do-while para hacer uso de esta estructura debemos conocer o tener declarada una bandera, ya que es hacer – mientras cierta bandera no este prendida, o cierta condición no se cumpla, esto resulta super útil para temas de login, ya que mientras el nombre del usuario o su contraseña no sean iguales esto se repetirá miles de veces; Como vemos igual que en las estructuras de selección todos tienen sus ventajas y desventajas, además de cual manejes mejor.

Práctica 10 la práctica se basó en lo que es la depuración de un código, que hace referencia a que cuando tenemos un código de un tamaño muy grande y el compilador nos da un error, esta opción nos dejara ver donde esta el error de una manera rápida y eficaz, facilita la corrección de un código, ya que si no tendríamos que ir buscando línea por línea investigando donde podría estar el error, esto se practicó con los ejercicios propuestos, y nos podíamos dar cuenta de como un punto y coma o un menor o mayor que se hayan confundido cambia toda la función que queríamos hacer en el programa.

Práctica 11 vimos lo que son las matrices, bueno el tema eran los arreglos bidimensionales como multidimensionales, entendimos que los arreglos en un primer caso nos servían para guardar varia información de un mismo tipo de carácter con el mismo nombre de variable, solo existiría un subíndice que determinaría que eran diferentes, estos al momento de declararlos teníamos que señalar la cantidad de memoria que queríamos reservar para ese arreglo; Otra función que nos da es el poder hacer matrices, teniendo la opción de llenarla con valores asignados o pidiéndoselos al usuario; Por otra parte a cada valor de un arreglo bidimensional como multidimensional, podemos acceder a él sin ningún problema para poder manipularlo, son super eficientes para almacenar nombres de algún grupo, prácticamente hacer listas, y los otros operaciones de matrices.

Práctica 12 funciones, consistió en elaborar varios programas en las cuales teníamos que recurrir a usar tanto variables locales como globales, con las locales eran las que declarábamos dentro de cada función y solo podíamos hacer uso de ella dentro de cada función, por otra parte, trabajamos las variables globales con las cuales podíamos hacer uso de ellas en cualquier momento, o hasta en otra función;

Práctica 13 fue la que mas se me complico en lo personal, ya que el tema se vio, pero no lo entendí perfectamente, la práctica consistió en hacer ejercicios relacionados a los archivos, ya que debíamos saber crear un archivo, leer los datos almacenados del archivo, agregar información al archivo, saber comparar información de un archivo, borrarlo; Bueno los archivos nos permiten almacenar información en un documento den blanco, dentro de la memoria de nuestro dispositivo, ya no solo a nivel de la consola del IDE, si no ya un archivo en el disco duro, esto se usaba con anterioridad para crear bases de datos, o llevar una organización de datos.