**Trabajo Practico AED**

**Mark Down:**

Markdown es una herramienta de conversion para programadores web, pasa texto a HTML. Markdown te permite escribir usando un formato de texto que es fácil de escribir y leer, y después se lo convierte a una estructura valida de XHTML(o HTML).

Markdown puede ser escrito en un editor de texto basico(no word). Es una forma simple de escribir texto que después puede ser traducido fácilmente a HTML. Las paginas web son escritas en HTML, entonces uno puede ver a esto como una herramienta de desarrollo web para los editores de contenido. Markdown es una forma de escribir contenido para la web.

A diferencia de otras aplicación de procesamiento de texto, el texto escrito por markdown puede ser compartido fácilmente entre computadoras, celulares y personas, por eso se esta convirtiendo en una herramienta básica para todo lo relacionado a la web. Paginas como github usan Markdown.

**Pseudocodigo:**

Mezcla de lenguaje de programación que se emplea, dentro de la programación estructurada, para realizar el diseño de un programa.

Se considera un primer borrador, dado que el pseudocódigo tiene que traducirse posteriormente a un lenguaje de programación.

Ventajas:

\_Ocupa poco espacio en el desarrollo de un problema.

\_Permite representar de forma fácil operaciones repetitivas complejas

\_Es mas sencilla la tarea de pasar de pseudocódigo a un lenguaje de progrmacion formal.

\_Se pueden observar claramente los niveles de estructura del programa

Desventajas:

\_Le cuesta mostrar flujos lógicos, estructuras de control.

\_La lógica usada en el pseudocódigo puede distar de la lógica usada por el programador.

\_Cada persona maneja su pseudocódigo.

**Diagrma de flujo:**

Un diagrama de flujo es la representación gráfica de un algoritmo.

Ventajas:

\_Favorecen la comprensión del proceso a través de mostrarlo como un dibujo.

\_Un buen diagrama de flujo reemplaza varias páginas de texto.

\_Permiten identificar los problemas y las oportunidades de mejora del proceso.

\_Es bastante sencillo y el más utilizado por su fácil comprensión y programación.

Desventajas:

\_Es el más ineficiente de todos los métodos.

\_No se elaboran con base en los principios de la programación estructurada, ilustran el flujo del programa, pero no su estructura.

\_Requiere de un espacio considerable en la hoja.

**Diagrama Nassi-Shneiderman:**

Basado en un diseño top-down (de lo complejo a lo simple), refleja la descomposición del problema en una forma simple usando cajas anidadas para representar cada uno de los subproblemas.

Ventajas:

\_Un enfoque mas estructurado, pero tal vez menos visual.

\_Adopta la filosofía de la programación estructurada, que utiliza un enfoque descendente. Utiliza un número limitado de símbolos de tal forma que el diagrama de flujo ocupa menos espacio. Puede leerse con cierta finalidad.

Desventajas:

\_Deben estar completos y ser muy claros, con el fin de que se entiendan.

\_Se utilizan muy raramente en las tareas de programación formal. Su nivel de abstracción es muy cercano al código de la programación estructurada y ciertas modificaciones requieren que se redibuje todo el diagrama.

**C++ :**

Es un lenguaje de programación diseñado a mediados de los años 1980 por Bjarne Stroustrup. La intención de su creación fue el extender al lenguaje de programación C mecanismos que permiten la manipulación de objetos.

Ventajas:

- Lenguaje de programación orientado a objetos.

- Lenguaje muy didáctico, gracias a este lenguaje puedes aprender muchos otros lenguajes con gran facilidad, como C#, Java, Visual Basic, Javascript, PHP, entre otros.

- Es muy potente en lo que se refiere a creación de sistemas complejos, un lenguaje muy robusto.

- Actualmente, puede compilar y ejecutar código de C, ya viene con librerías para realizar esta labor.

Desventajas:

\_Es mucho más difícil de usar q otros lenguajes de programación.   
\_ No es recomendable para desarrollo de páginas Web.

**Conclusion:**

Programar en c++ directo no permite a uno abstraerse y ver el panorama completo del programa. Entonces como buenas medidas para abstraernos tenemos pseudocódigo que es el mas similar a la programacion real, pero peor visualización del programa. Para eso tenemos el diagrama de flujo que permite ver el algoritmo de forma mas grafica y comprensible pero si el programa es extenso y/o complejo se hace tedioso e ineficiente el diagrama de flujo y en ese caso podemos recurrir a pseudo o Nassi-Shneiderman. En el caso de Nassi-Shneiderman tenes que aprender los símbolos que usa, que puede ser un poco complejo, pero una vez aprendido resulta de mucha facilidad la lectura del programa junto con todos sus procesos.