**Instituto Superior de Formación Técnica Nº 151   
Carrera: Analista en Sistemas  
1 Año. Algoritmos y Estructuras de Datos.**

|  |  |
| --- | --- |
| **Trabajo Práctico Nº1** | **Unidad 1** |
| **Modalidad:** Semi -Presencial | **Estratégica Didáctica:** Trabajo individual. |
| **Metodología de Desarrollo:** Det. docente | **Metodología de Corrección:** Via Classroom. |
| **Carácter de Trabajo:** Obligatorio – Con Nota | **Fecha Entrega:** A confirmar por el Docente. |

**Marco Teórico:**  
Responder la siguiente cuestionario en función de la bibliografía Obligatoria.

1. Definir los conceptos de Informática y computadora.
2. Definir Hardware y Software
3. Describir los roles de los Programadores
4. Relacionar los conceptos de la Diapositiva 9
5. Qué entiende por lenguaje de Máquina
6. Qué entiende por lenguaje Ensamblador
7. Que facilitan los Lenguajes de Alto nivel y explicar el ciclo de Compilación.
8. Definir el concepto de Programa Informático
9. Describir el Modelo en Cascada
10. Buscar Información sobre el sistema Evolutivo e Incremental.

La Cátedra.

Lic. Oemig José Luis.

1. Informática: Conjunto de conocimientos científicos y técnicas que hacen posible el tratamiento automático de la información por medio de ordenadores.

Computadora: Máquina electrónica, analógica o digital, dotada de una memoria de gran capacidad y de métodos de tratamiento de la información, capaz de resolver problemas matemáticos y lógicos mediante la ejecución de programas informáticos

1. Hardware: Componentes que integran la parte material de una computadora.

Software: Programas, instrucciones y reglas informáticas para ejecutar tareas en una computadora

1. Roles de los Programadores:

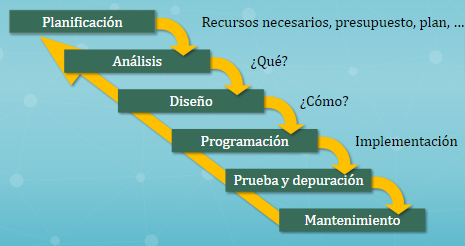
Los programadores, además de trabajar en equipo, deben cumplir múltiples roles:

* Gestores
* Analistas
* Diseñadores
* Programadores
* Probadores
* Administradores de Sistemas…

1. Una computadora se compone de las siguientes partes:

Una CPU (Unidad Central de Proceso), la cual se encuentra conectada a una Memoria Temporal (allí se almacenan datos para un rápido acceso, pero dichos datos se pierden al apagar la computadora), una Unidad de Almacenamiento Permanente (el acceso a los datos allí almacenados no es tan veloz como en la Memoria Temporal dichos datos persisten aún cuando el ordenador se apaga), Dispositivos de Entrada (ratón, teclado, escáner, pantallas táctiles, etc. todos dispositivos mediante los cuales introducimos datos), Dispositivos de Salida (monitor, impresora, altavoces, etc. dispositivos de salida de datos)

1. Lenguaje de Máquina: Lenguaje conformado por instrucciones en código binario, usado directamente por la computadora.
2. Lenguaje Ensamblador: Lenguaje de programación de bajo nivel en el que se emplea para representar cada una de las instrucciones del lenguaje de máquina para una computadora particular.
3. Los lenguajes de Alto Nivel son más cercanos al lenguaje natural y matemático y permiten una mayor facilidad de uso que un Lenguaje Ensamblador. Un programa llamado *compilador* traduce los programas (**programa fuente**) escritos en algunos lenguajes de alto nivel (C++, Pascal, FORTRAN, COBOL, Modula-2 y Ada, por ejemplo) en lenguaje de máquina (**programa objeto**).
4. Un Programa Informático no es más que la implementación de un algoritmo en un lenguaje concreto. El mismo transforma una entrada en una salida.
5. Modelo de desarrollo “en cascada”:



Para desarrollar software convenientemente, se debe **Planificar**, esto es: identificar los recursos necesarios para la concreción de la tarea, presupuestar todo el proyecto y tener un plan de avance del mismo. Luego se debe realizar un **Análisis** pormenorizado tanto del problema como de las posibles soluciones. En base a la solución elegida, se realizará el **Diseño** del algoritmo. Como paso siguiente se llevará a cabo la **Programación** de dicho algoritmo. El paso siguiente será la Implementación del mismo, donde se harán **Pruebas**  se **Depurará** el código. A esto le sigue el **Mantenimiento** del Software. Aquí se deberá hacer un seguimiento del desempeño del mismo y, en caso de merecer mejoras o actualizaciones, se deberán repetir los pasos mencionados.

1. Sistema Evolutivo e Incremental: es un proceso de desarrollo de software creado en respuesta a las debilidades del modelo tradicional de cascada.

Básicamente este modelo de desarrollo, que no es más que un conjunto de tareas agrupadas en pequeñas etapas repetitivas (iteraciones) es uno de los más utilizados en los últimos tiempos ya que, como se relaciona con novedosas estrategias de desarrollo de software y una programación extrema, es empleado en metodologías diversas.

El modelo consta de diversas etapas de desarrollo en cada incremento, las cuales inician con el análisis y finalizan con la instauración y aprobación del sistema.

El modelo de gestión incremental no es un modelo necesariamente rígido, es decir, que puede adaptarse a las características de cualquier tipo de proyecto, existen al menos 7 fases que debemos tener en cuenta a la hora de implementarlo:

**Requerimientos**: son los objetivos centrales y específicos que persigue el proyecto.

**Definición de las tareas y las iteraciones**: teniendo en cuenta lo que se busca, el siguiente paso es hacer una lista de tareas y agruparlas en las iteraciones que tendrá el proyecto. Esta agrupación no puede ser aleatoria. Cada una debe perseguir objetivos específicos que la definan como tal.

**Diseño de los incrementos**: establecidas las iteraciones, es preciso definir cuál será la evolución del producto en cada una de ellas. Cada iteración debe superar a la que le ha precedido. Esto es lo que se denomina incremento.

**Desarrollo del incremento**: posteriormente se realizan las tareas previstas y se desarrollan los incrementos establecidos en la etapa anterior.

**Validación de incrementos**: al término de cada iteración, los responsables de la gestión del proyecto deben dar por buenos los incrementos que cada una de ellas ha arrojado. Si no son los esperados o si ha habido algún retroceso, es necesario volver la vista atrás y buscar las causas de ello.

**Integración de incrementos**: una vez son validados, los incrementos dan forma a lo que se denomina línea incremental o evolución del proyecto en su conjunto. Cada incremento ha contribuido al resultado final.

**Entrega del producto**: cuando el producto en su conjunto ha sido validado y se confirma su correspondencia con los objetivos iniciales, se procede a su entrega final.