



Laboratorio de Computación I

Prof. Nelson Benaventana
Ingeniero en Sistemas

La carrera

Que es la Tecnicatura Superior en Programación?

- Analizar problemas de procesamiento de datos.
- Desarrollarlos.
- Depurarlos.
- Transferirlos.
- Diseñar e implementar los sistemas requeridos por el mercado laboral.
- Optimizar los procesos acorde a las necesidades que impone la competencia del mercado.

Para que nos servirá estudiar esta carrera?

El mercado laboral tiene la necesidad de cubrir la falta de personal técnico capacitado en el "Área Informática", ya que el mundo presenta una rápida evolución y que el personal administrativo contable, hoy por hoy, necesita de conocimientos básicos de programación. (Análisis de datos, Manejo de Software, Etc.)

Que aprenderemos a hacer?

La capacitación, a través de la carrera, le permitirá colaborar en el planeamiento, desarrollo, dirección, implementación y control de proyectos en el ámbito de la Tecnología Informática.

► Porque vamos a estudiar esto?



La materia

La asignatura está orientada a la comprensión de las interfaces de usuarios y su diseño, como así también a la introducción al alumno en un entorno de desarrollo en equipo, para que comience a adquirir la base de la programación. Esta asignatura también infundirá las buenas prácticas en la programación y la calidad en la programación de aplicaciones.

La materia

Plan de Unidades

UNIDAD 1: Introducción a la programación.

UNIDAD 2: Programación Visual en Visual Basic 6.

UNIDAD 3: Diseño de Interfaces de Usuarios.

UNIDAD 4: Manejo y control de errores.

UNIDAD 5: Documentación y depuración de Programas.

UNIDAD 6: Tecnología .Net

La materia

Condiciones de:

Promoción: Cumplir con la asistencia del 75% a las clases teóricas y prácticas y aprobar el 100% de los parciales con una calificación de 7 (siete) a 10 (diez) puntos.-

Regularización: Asistencia del 75% a las clases teóricas y prácticas. Aprobar el 100% de los parciales con una calificación mínima de 6 (seis) puntos.-

Para ambos casos, cumplir con la entrega de la totalidad de los TP.

Fecha de Parciales y recuperatorios (Agendar**):**

✓1^{er} Parcial: 03/05/2021 – Recuperatorio: 10/05/2021

✓2^{do} Parcial: 14/06/2021 – Recuperatorio: 21/06/2021

Primer día

1- Presentación de los alumnos.

- Nombre y apellido
- Ocupación.

2- Cuestionario de nivelación.

Primer día

Cuestionario de nivelación

¿Qué es un sistema operativo? Nombre algunos.

¿Qué es un programa?

¿Qué programas conoce? Nómbrelos.

¿Qué programas utiliza a menudo?

¿Qué actividades realiza con los mismos?

¿Qué son los Datos?

¿Qué tipos de datos conoce?

¿Programó alguna vez?

¿Conoce algún lenguaje de programación? ¿Lo ha utilizado?

¿Su actividad diaria está ligada fuertemente con la informática? Comente.

Sistema Operativo

► El programa fundamental de todos los programas de sistema, es el Sistema Operativo, que controla todos los recursos de la computadora y proporciona la base sobre la cual pueden escribirse los programas de aplicación.



► Un Sistema Operativo es un programa que actúa como intermediario entre el usuario y el hardware de un computador y su propósito es proporcionar un entorno en el cual el usuario pueda ejecutar programas y realizar sus tareas

► El objetivo principal de un Sistema Operativo es, entonces, lograr que el Sistema de computación se use de manera cómoda, y el objetivo secundario es que el hardware del computador se emplee de manera eficiente.

Sistema Operativo

► Definición:

Es un conjunto de módulos o funciones (software) que se ocupan de controlar y administrar la ejecución de los programas sobre los recursos que brinda el equipo (hardware) tales como: memoria, procesador, periféricos, etc.



Algunos Sistemas Operativos



Programa o aplicación

► Un programa es un conjunto de instrucciones u ordenes basadas en un lenguaje de programación que una computadora interpreta para resolver un problema o una función específica.



Lenguaje de programación

► Un conjunto de sintaxis y reglas semánticas que definen los programas de la computadora. Es una técnica estándar de comunicación para entregarle instrucciones a la computadora.

► Un lenguaje le da la capacidad al programador de especificarle a la computadora, qué tipo de datos actúan y que acciones tomar bajo una variada gama de circunstancias, utilizando un lenguaje relativamente próximo al lenguaje humano.

Lenguaje de programación

- Los lenguajes de programación pueden ser clasificados de acuerdo a diversos criterios:
- **Lenguajes interpretados** (Interpretes) : Basic, Dbase.
- **Lenguajes compilados** (Compiladores) : C, C++, Clipper.
- **Lenguajes interpretados con recolectores de basura** (Maquina Virtual) como Smalltalk, Java.
- **Lenguajes Scripts** (Motor de ejecución) : Perl, PhP.



Unidad 1:

Comenzar a programar

- **Análisis del programa**
- **Algoritmos**
- **Algoritmos en Diagramas de Flujos de Datos**
- **Algoritmos en Pseudocódigo**

Comenzar a programar

- **Análisis del programa**
- Analizar los objetivos (**qué debe hacer el programa**) y los pasos necesarios para alcanzarlos, creando así un programa robusto (libre de errores).

Comenzar a programar

- **Algoritmos**

- Una vez analizados los requisitos del programa podemos crear algoritmos para indicar qué pasos debe realizar el software de modo que realice correctamente su propósito, independientemente del lenguaje de programación que sea usado.

- Podemos representar los algoritmos de 3 modos:

- Usando lenguaje natural

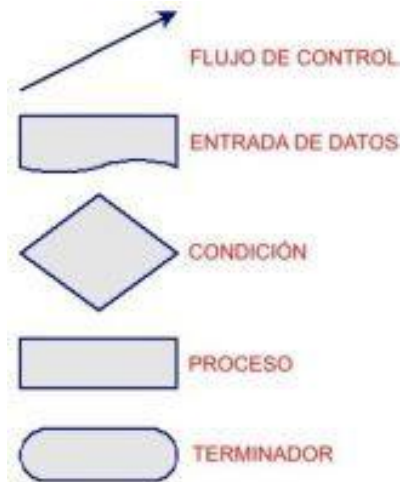
- Pseudocódigo

- Diagramas de Flujo de Datos (DFD)

- Para el presente curso nos centraremos principalmente en el pseudocódigo.

Comenzar a programar

- **Algoritmos en Diagramas de Flujos de Datos (DFD)**
- Para crear un DFD usaremos entre otros los siguientes símbolos:



Ejemplo de un programa que pide un número al usuario y muestra el resultado por pantalla, realizado con un Diagrama de Flujo de Datos (DFD):



Comenzar a programar

- **Algoritmos en pseudocódigo**

- El pseudocódigo es una mezcla entre lenguaje natural y lenguaje de programación. La estructura básica de un algoritmo en pseudocódigo es:

1. ALGORITMO Mostrar;

2....

3. INICIO

4. // Esto es un comentario

5. /*

6. Esto es un comentario

7. Esto es otra línea dentro del mismo comentario

8. */

9. ...

10. FIN

Comenzar a programar

- **Tipos de datos**

- Los principales tipos de datos usados en los lenguajes de programación son:

- **Caracter (String)**: para almacenar caracteres y cadenas de texto. En algoritmos en pseudocódigo se declaran con la palabra CHARACTER.

- **Lógicos o booleanos**: del tipo Verdadero/Falso. En pseudocódigo se declaran con la palabra BOOLEAN.

- **Numéricos**: almacenan números sin decimales (ENTERO) o con decimales (REAL).

Comenzar a programar

- **Identificadores**

- Se denomina como identificador al nombre que podemos asignar a determinados elementos en el código fuente del programa, como por ejemplo a las **variables**, **funciones**, etc.

- Para crear un identificador debemos tener en cuenta las siguientes reglas:

- Debe comenzar por una letra, aunque el resto del nombre puede contener también números.

- Se pueden usar tanto mayúsculas como minúsculas, dependiendo del lenguaje de programación usado, se hace distinción entre ambas. Serían diferentes tres elementos llamados 'Comision', 'comision' y 'COMISION'.

- No se pueden usar tildes.

- También es posible usar el carácter de subrayado '_'.

- Algunos identificadores válidos: 'x', 'el_nombre', 'salario', 'mes1'...

- Algunos identificadores no válidos: '3po' (comienza con un número), '3+4' (contiene un carácter no válido), 'comisión' (no se admiten tildes).

Comenzar a programar

- **Variables y constantes**

- En cualquier lenguaje de programación necesitaremos usar **variables** y **constantes** para almacenar determinados valores (los cuales se guardan internamente en una posición de memoria) y así poderlos recuperar posteriormente para usarlos en otra parte del código fuente.

- La diferencia entre ellas es que a las **constantes** se les puede asignar su valor una sola vez, mientras que el valor de las **variables** puede ser modificado en cualquier punto del código.

Comenzar a programar

• Variables y constantes

1. ALGORITMO Variables1;

2. CONS

3. PI <- 3,14;

4. VAR

5. ENTERO num1;

6. ENTERO num2 <- 7;

7. CARACTER nombre, apellidos;

8. BOOLEAN esta_bien <- ok;

9. INICIO

10. num1 <- 5;

11. nombre <- "PEPE";

12. apellidos <- "PÉREZ PÉREZ";

13. FIN

14. Para asignar un valor en pseudocódigo usamos ' \leftarrow ', aunque en otras nomenclaturas se usa el signo igual '=' (que es usado también en los lenguajes de programación).

Comenzar a programar

- **Variables y constantes**

1. ALGORITMO Variables2;

2. VAR

3. CHARACTER nombre;

4. INICIO

5. LEER(nombre);

6. ESCRIBIR("Hola ", nombre);

7. FIN

8. En pseudocódigo se usa LEER() para solicitar un dato introducido por teclado, y ESCRIBIR() para mostrar un texto por pantalla.

Comenzar a programar

- **Resolver**

1. Elaborar un algoritmo que solicite el numero de respuestas correctas, incorrectas y en blanco, correspondientes a postulantes de un concurso de trabajo y muestre su puntaje final considerando que por:

1. Cada respuesta correcta tendrá 4 puntos.

2. Respuestas incorrectas -1 punto.

3. Respuestas en blanco 0 puntos.

Comenzar a programar

- **Resolver**

1. Elaborar un algoritmo que permita cargar la cantidad de partidos, ganados, perdidos y empatados de cualquier equipo del campeonato y calcule el puntaje final teniendo en cuenta que:

1. Partido ganado suma 3 puntos.

2. Partido perdido suma 0 puntos.

3. Partido empatado suma 1 punto.

Comenzar a programar

Operadores matemáticos

+ (Suma)

- (Resta)

* (Multiplicación)

/ (División)

^ (Potencia)

MOD (Resto de la división entera)

SQRT (Raiz Cuadrada)

ABS (Valor absoluto)

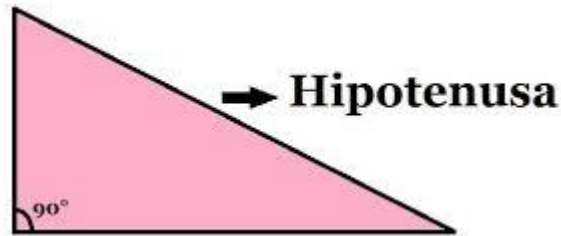
TRUNC (Parte entera)

RANDOM (Número aleatorio)

Comenzar a programar

Resolver

Elaborar un algoritmo que me permita ingresar los valores de los catetos de un triángulo rectángulo y me calcule el valor de la hipotenusa.



Unidad 2

Programación en VB 6

Comenzaremos a trabajar en Visual Basic 6.