

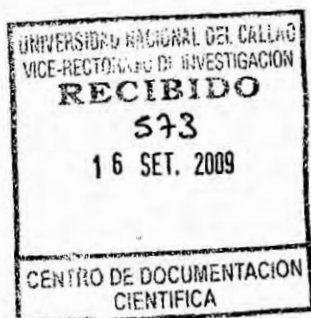


OCT. 2009

UNIVERSIDAD NACIONAL DEL CALLAO

FACULTAD DE INGENIERIA QUIMICA

INSTITUTO DE INVESTIGACION



**“TEXTO UNIVERSITARIO: PROBLEMAS EN VISUAL
BASIC.NET APLICADOS A LA INGENIERÍA QUÍMICA”**

(Informe Final)

LIC. SALVADOR APOLINAR TRUJILLO PEREZ

(01/10/07 AL 30/09/09)

(Resolución Rectoral N° 1214-2007-R)

CALLAO-PERU

2009



INDICE

1. RESUMEN.....	3
2. INTRODUCCION.....	4
2.1. Presentación del Problema de Investigación.....	5
2.2. Planteamiento del Problema de Investigación.....	5
2.3. Objetivo de la investigación.....	6
2.4. Importancia y justificación de la Investigación.....	7
2.5. Enunciado de la Hipótesis.....	7
3. MARCO TEORICO.....	8
3.1. Estructura de un programa.....	8
3.2. Sentencias de control.....	8
3.2.1. IF THEN	
3.2.2. IF THEN ELSE	
3.2.3. SELECT CASE	
3.2.4. DO WHILE / UNTIL LOPP	
3.2.5. WHILE	
3.2.6. FOR NEXT	
3.3. Arreglos Unidimensional.....	9
3.4. Arreglos Bidimensionales (Matrices).....	9
3.5. Procedimiento.....	10
3.6. Funciones.....	10
3.7. Problemas propuestos.....	10
4. MATERIALES Y METODOS.....	11
4.1. Materiales.....	11
4.2. Métodos.....	11

5. RESULTADOS.....	12
6. DISCUSION.....	13
7. REFERENCIALES.....	14
8. APENDICES.	16
ANEXOS.....	25



1. RESUMEN

El propósito del presente trabajo de investigación fue la elaboración de un "Texto Universitario: Problemas en Visual Basic.net aplicados a la Ingeniería Química" que sirva de complemento a la formación que se brinda en el aula y como una orientación en el estudio y reforzamiento de las enseñanzas brindadas por el profesor, lo cual significará un valioso aporte para los estudiantes de ciencias e ingeniería y en particular para los alumnos de la Facultad de Ingeniería Química de la Universidad Nacional del Callao.

La metodología utilizada para la elaboración, se sustenta en la revisión bibliográfica y la experiencia del autor como profesor de los cursos de Programación de Computadoras para Ingeniería. La didáctica que el autor ha venido utilizando y brindando al los estudiantes durante muchos años, ha sido mejorado a través del tiempo incorporando cada vez temas de actualizados sobre la materia, lo que redundará en el beneficio de los alumnos que cursan dicha asignatura y que ha permitido al autor definir el contenido del presente texto

En general se ha logrado elaborar un texto sencillo y práctico, de fácil entendimiento y dirigido a los estudiantes que inicien el curso Programación de Computadoras para Ingeniería.

Se espera que la presentación y la metodología utilizada realmente representen utilidad en el proceso de enseñanza aprendizaje de la Programación de Computadoras.

2. INTRODUCCION

El tema, materia de la investigación, es el desarrollo de un "Texto Universitario Problemas en Visual Basic.net Aplicados a la Ingeniería Química", dirigido a los estudiantes universitarios del pre-grado en Ingeniería Química, se presentan los diversos **problemas** a los puntos principales de la programación en Visual Basic, como las sentencias de control para el cual se utilizan las sentencias If Then, If Then Else, Select Case, Do While/ Until Loop, For next, elementos básicos del Lenguaje de programación Visual Basic.

Los métodos y técnicas para el desarrollo de programas que cumplan con ciertos requisitos de calidad. Y también las implicaciones que la relaciona ciencia – tecnología que le permitirá cumplir con los propósitos de una adecuada enseñanza y formación profesional, se presentan problemas Químicos que para facilitar la solución se utiliza un lenguaje de programación, en este caso es el Visual Basic.

La principal razón para que los alumnos aprendan lenguajes y técnicas de programación es utilizar la computadora como una herramienta para resolver problemas.

El texto desarrollado y elaborado por el autor tiene una estructura sencilla, la cual pretende ser un instrumento práctico de fácil entendimiento, diseñado específicamente para los alumnos del pre-grado de la carrera de Ingeniería Química.

2.1 Presentación del Problema de Investigación

El tema de Investigación a desarrollar, es la elaboración de un Texto Universitario Titulado "Problemas en Visual.Net aplicados a la ingeniería Química"

En el medio no hay textos Universitarios similares al propuesto. Los textos que tratan temas similares al propuesto, están dirigidos al campo de las matemáticas, los alumnos encuentran dificultades de poder programar en Visual Basic con aplicaciones a la Ingeniería Química, debido a la necesidad del alumno de una orientación personalizada. Si bien el Texto Universitario propuesto, no cumplirá con la necesidad expuesta, si contribuirá con dicha necesidad.

Durante los años que tengo a mi cargo el desarrollo de este tema, se ha observado que los estudiantes carecen de material bibliográfico suficiente, con esta clase de detalles, que le permita un aprendizaje adecuado.

2.2 Planteamiento del problema de Investigación

¿Cómo lograr un Texto Universitario que facilite la resolución de Problemas en Visual Basic.net con aplicaciones a la ingeniería Química?

2.3 Objetivo de la Investigación

a) OBJETIVOS GENERALES

Desarrollar un texto universitario, que facilite la Construcción de Problemas y su respectiva Programación de Computadoras utilizando el lenguaje Visual Basic.Net con aplicaciones a la Ingeniería Química.

b) OBJETIVOS ESPECIFICOS

- i. Acopiar información básica y actualizada, necesaria para iniciar el desarrollo del Texto.
- ii. Analizar y procesar la información básica acopiada para iniciar el desarrollo del texto.
- iii. El área de aplicación es la Química General y los temas a tratar son: Gases, Equilibrio Químico, Equilibrio Acido Básico, Cinética Química, Química de la Atmosfera, etc.
- iv. Desarrollar los capítulos del texto, referido a enunciados de Problemas y su codificación en Visual basic.net
- v. Elaborar el informe Final de la Investigación realizada.

2.4 Importancia y Justificación de la investigación

El Texto Universitario que se propone desarrollar, pretende ser un medio complementario a la formación que se dan en las aulas: algunas veces los alumnos no captan la enseñanzas en clase, por varias razones, dentro las cuales están el no haber asistido a clase, el haber llegado tarde o si están presentes se distraen por alguna preocupación

El texto universitario que se desarrollara como tema de Investigación, significara un valioso aporte para los alumnos y egresados de Ingeniería Química, por ser un medio de orientación complementario a la formación que se dan en las aulas. En nuestro medio no hay texto universitario de programación en Visual Basic que tengan aplicaciones a la Ingeniería química. Los textos que existen relacionados al tema lo hacen en los campos de las matemáticas.

2.5 Enunciado de la Hipótesis

El texto "Problemas en Visual Basic.net aplicados a la Ingeniería Química", que se desarrollara; explicara como adecuar el método científico en las investigaciones relacionadas a la Ingeniería Química y lo que conviene Investigar al egresado o alumno de esta especialidad.

3. MARCO TEORICO

La estructura del "Texto Universitario Problemas en Visual Basic.net Aplicados a la Ingeniería Química", se hace en orden creciente de Complejidad dentro de cada tema y se trata de lograr un ordenamiento de los mismos de acuerdo con las necesidades de conceptos para ir avanzando en la proposición de ejercicios.

A tal fin se propone la siguiente estructura

3.1 Estructura de un programa.

El concepto de programa como conjunto de instrucciones y sus tipos constituye la parte fundamental del capítulo. La descripción de los elementos básicos de programación, que se encontraran en casi todos los programas, contadores, acumuladores, etc.

Se denomina datos a las características propias de cualquier entidad. Cada variable, constante o expresión lleva asociado un tipo de dato que determina un conjunto de valores que puede tomar.

3.2 Sentencias de control.

En este módulo, aprenderemos a:

Resolver problemas en Visual Basic aplicados a la ingeniería química, creando fórmulas y expresiones condicionales utilizando operadores aritméticos, de comparación y lógicos.

- 3.2.1 Utilizar instrucciones **If...Then** para evaluar si una condición es verdadera y dirigir el flujo del programa en consecuencia.

- 3.2.2 Utilizar instrucciones **If...Then Else** para evaluar si una condición es verdadera o falsa y dirigir el flujo del programa en consecuencia.
- 3.2.3 Utilizar instrucciones **Elseif** para evaluar si una serie de condiciones anidadas y dirigir el flujo del programa en consecuencia.
- 3.2.4 Utilizar instrucciones **Select Case** para probar diferentes valores de la misma expresión y ejecutar las instrucciones correspondientes
- 3.2.5 Utilizar instrucciones **Do...Loop** para ejecutar instrucciones hasta o mientras se satisface una condición específica
- 3.2.6 Utilizar instrucciones **For...Next** para ejecutar instrucciones un determinado número de veces

3.3 Arreglos Unidimensionales

Un Array (matriz o vector) es un conjunto finito y ordenado de elementos homogéneos. En el cual se resolverá una variedad de problemas aplicados a la ingeniería química.

3.4 Arreglos Bidimensionales (Matrices)

Un array bidimensional se puede considerar como un vector de vectores, por consiguiente un conjunto de elementos, todos del mismo tipo, en el cual el orden de los componentes es significativo y en el que se necesita especificar dos subíndices para poder identificar cada elemento del array. En este capítulo se tiene la particularidad de resolver los problemas en visual Basic, trabajando en modo Consola.



3.5 Procedimiento

Los procedimientos son las sentencias de código ejecutable de un programa.

Las instrucciones de un procedimiento están delimitadas por una instrucción de declaración y una instrucción **End**.

3.6 Funciones

Un procedimiento **Function** es una serie de instrucciones Visual Basic delimitadas por las instrucciones **Function** y **End Function**. Los procedimientos **Function** son similares a los procedimientos **Sub**, pero las funciones pueden devolver un valor al programa que origina la llamada.

3.7 Problemas propuestos Aplicados a la Ingeniería Química.

En este capítulo proponemos una serie de problemas químicos, es decir planteamos el problema, diseñamos el Formulario y definimos todas las variables que se van a utilizar.

4 MATERIALES Y METODOS

4.5 Materiales

- Materiales de oficina
- Materiales de consulta
- Materiales de computo e impresiones

4.6 Métodos

La elaboración del texto, propósito de la investigación, ha demandado al autor el ordenamiento de la información compilada durante su vida profesional y académica, al desempeñarse primero como profesor titular del curso de Programación de Computadoras para Ingeniería desde el año 1999 hasta la presente fecha.

Ser profesor del curso de Programación de Computadoras para Ingeniería, en la Facultad de Ingeniería Química, de la Universidad Nacional del Callao, le dio la oportunidad al autor de ir elaborando separatas de los capítulos del presente texto y poder comprobar con los alumnos, determinadas necesidades de modificación, hasta tener los mejores resultados de entendimiento por parte de los alumnos.

La experiencia adquirida, durante el periodo de trabajo en la UNAC ha contribuido a lograr el texto con las características didácticas que se presentan.

5 RESULTADOS

El resultado de la presente investigación es el texto universitario que se adjunta, titulado: "Texto Universitario: Problemas en Visual Basic.net Aplicados a la Ingeniería Química ", El texto elaborado contiene siete capítulos, expuestos en forma práctica que permite una fácil y rápida interpretación de los alumnos del pre-grado del área de Ingeniería Química

El texto presenta problemas resueltos y propuestos aplicados a la Ingeniería química y se presenta con sencillez y de tal manera que el alumno logre entenderlo y manejarlos en forma adecuada

Para cada capítulo se hace una descripción de los principio, aplicaciones, cálculos y relaciones con otros temas ya desarrollados y también una variedad de problemas resueltos aplicados a la ingeniería química, con su diseño de Formulario y su respectiva codificación en Visual Basic, y en un capítulo final se presenta un número de problemas propuestos de varios tipos y niveles de dificultad.

6 DISCUSION

El texto universitario, titulado "Texto Universitario: Problemas en Visual Basic.net aplicados a la Ingeniería Química" que es el resultado de la presente investigación se caracteriza por presentar en una forma ordenada, sencilla y de fácil comprensión, una variedad de problemas resueltos, propuestos, y aplicados a la Ingeniería química de manera que el alumno logre entenderlo y manejarlo en forma adecuada.

Si bien existen algunos textos de nivel universitario que se usan en los claustros universitarios, la mayoría de estos textos o libros de Programación de Computadoras presentan los ejercicios están orientados a problemas básicamente de orientación matemática y no a la ingeniería química

Este texto intenta superar esto con problemas resueltos y aplicados a la Ingeniería Química, que ha sido desarrollado mediante un cuidadoso estudio, tiene la intención de presentar al lector de una manera sencilla y practica el uso, y solución de problemas Químicos en Visual Basic.net.

UNIVERSIDAD NACIONAL DEL CALLAO

FACULTAD DE INGENIERIA QUIMICA

SYLLABUS

1. Información General

- 1.1 Nombre Asignatura: Programación de Computadoras para Ingeniería
- 1.2 Código : IG102
- 1.3 Carácter : Obligatorio
- 1.4 Crédito : 03
- 1.5 Horas : Teoría 02 ; Laboratorio: 03
- 1.6 Ciclo Académico : **2009-B**
- 1.7 Duración : 17 Semanas
- 1.8 Pre-requisito : FM102
- 1.9 Profesor : Salvador Trujillo Pérez

Grupo Horario 01Q : Lic. Salvador Trujillo Pérez

Grupo Horario 02Q : Lic. Salvador Trujillo Pérez

Laboratorio 90G : Lic. Jeanete Estrada Cantero

Laboratorio 91G : Lic. Salvador Trujillo Pérez

Laboratorio 92G : Lic. Salvador Trujillo Pérez

Laboratorio 93G : Lic. Jeanete Estrada Cantero



2.- Objetivos.

Generales.

- Dotar al estudiante de los conocimientos básicos de la informática y la programación de computadoras
- Crea en los estudiantes el hábito del uso del avance de la ciencia informática relacionada con la ingeniería Química.

Específicos

- Proporcionar una visión sistemática de la informática actual en el siglo XXI e Internet
- Analizar un problema, buscar algoritmos matemáticos para la solución del problema.
- Compilar, Ejecutar y catalogar programas con el lenguaje de Programación Científico (Visual Basic.net).

3.- Sumilla.

En el curso se tratará los siguientes temas. Introducción a la informática. Algoritmos y diagramas de flujo. Estructura del lenguaje de programación. Sentencias aritméticas y de control arreglos: Vectores y Matrices. Procedimientos. Funciones. Tratamientos de cadenas de caracteres. Estructura de datos. Gráficos. Programación orientada a objetos. Paquetes aplicados a la ingeniería Química.

4.- Metodología

El desarrollo de la asignatura se efectuara sobre la base de los siguientes lineamientos didácticos.

Las clases serán teóricas practicas, desarrollándose los temas de acuerdo al programa analítico diseñado. El profesor propiciara y estimulara la participación de los alumnos de clase.

El alumno deberá asistir a la clase obligatoriamente, estudiando los temas tratados y repasando el tema que el profesor desarrollara. Esto permitirá una mejor participación. del alumno en clase.

El profesor de la asignatura brindara horas de asesoría en horarios predeterminados con el fin de atender en forma personalizada., cualquier dificultad que el alumno pudiese encontrar en el estudio de los distintos tópicos.

El profesor pondrá a disposición de los estudiantes separatas y guías de prácticas, que deberán ser resueltos con finalidad de afianzar los conocimientos adquiridos.

5. Contenido Analítico y calendarización.

1 semana Elementos y conceptos Fundamentales: Hardware. Software. Bit. Byte. Megabyte, Gigabyte, Tetra byte. Evolución de las computadoras. Capacidad de memoria RAM, CD, Hard disk y disquetes. Informática en el siglo XXI. Internet.

2 semana Algoritmo y diagrama de flujo. Símbolos de operación, símbolos de decisión, líneas de flujo. Ejercicios de diagramación y programación

3 semana Estructura de un lenguaje de Programación y elementos, objetos de un programa. Identificadores. Tipos de datos, constantes y variables.

4 semana Expresiones. Tipos de expresiones. Operadores. Tabla de verdad de los operadores lógicos, orden de evaluación de los operadores.

5 semana Sentencias Aritméticas: sentencias de entrada / salida. Ejemplos variados de programas.

6 semana Sentencias de control. If Then - IF Then Else, While, Select Case, do Loop, For Next.

7 semana Arreglos Unidimensionales (Vectores)

8 semana **Primer examen Parcial.**

9 semana Arreglos Bidimensionales (Matrices).

10 semana Procedimientos. Declaración y llamada, variables locales y globales, paso de parámetros. Procedimientos anidados.

11 semana Funciones Creación y llamadas a sub programas.

12 semana Tratamientos de cadenas de caracteres. String, operaciones entre cadenas, procedimientos y funciones de cadenas internas.

13 semana Estructura de datos. La declaración Type, conjuntos, pertenencia, operaciones con conjuntos, registros, manejo de los campos de un registro la sentencia With.

14 semana Gráficos. La unidad Graph, inicialización del controlador de gráficos, conmutación entre modo texto y modo grafico, control de colores en gráficos, dibujo de líneas y figuras, visualización de texto en pantalla de gráficos.

15 semana Paquetes aplicados a la Ingeniería Química.

16 semana **Segundo examen Parcial**

17 semana **Examen sustitutorio.**

6.- Evaluación

- La evaluación del rendimiento de los alumnos es objetiva, sobre la base de: Prácticas calificadas, un examen Parcial y un examen final e intervenciones orales.
- Si la nota final fuese desaprobadora, el alumno rendirá un examen sustitutorio, que será el único y abarcará todo el curso. Reemplaza a la nota más baja de los exámenes.
- La nota final será obtenida por el siguiente modo:

$$\text{Nota Final} = \frac{PP + E1 + E2}{3}$$

Donde:

PP : Es el promedio de prácticas calificadas incluido las intervenciones orales.

E1 : Es el primer examen parcial.

E2 : Es el segundo examen parcial.

7.- Equipos Materiales.

En el desarrollo del curso los siguientes materiales y equipos.


- Tizas, papel, separatas, lapiceros etc.
- Computadoras para el desarrollo de las prácticas.

8.- Bibliografía.

1. Cristian Sánchez Flores Desarrollando aplicaciones con
Visual Basic 2008
Editora Macro,
1ra, Edición octubre 2008-Peru
2. Francisco Charte Ojeda Programación Visual Basic 2008
Editora Anaya
2009, Madrid
3. Evangelos Petroustos La Biblia
Visual Basic 2008
Editora Anaya- 2009
4. Peter Norton Introducción a la Computación
Mc Graw-Hill
Interamericana, 2da. Edición, México.
5. Osvaldo Cairo Battistutti Metodología de la Programación tomo I y II
(Algoritmos, diagramas de flujo y programas)
1995

6. Eduardo Alcalde-M. García Metodología de la Programación.

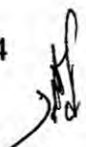
7. Julio Vasquez Paragulla Diseño de Programación
(Pseudocódigo, Diagrama de Flujo)
3era Edición – 1997



.....
Profesor: Salvador Trujillo Pérez

Lic. En Investigación Operativa

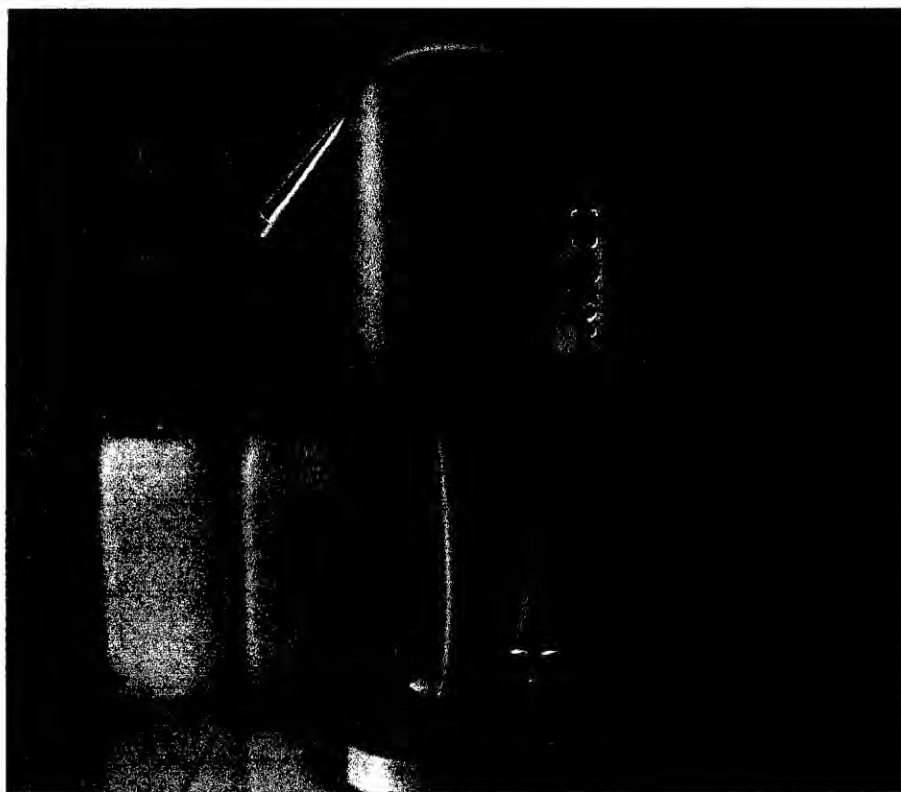
Callao, Agosto del 2009



9 ANEXOS

Texto:

Problemas en Visual Basic.net



Aplicados a la Ing. Química

*Lic. Salvador Trujillo Pérez
Profesor de la Facultad de Ingeniería
Química
Universidad Nacional del Callao
Setiembre 2009*

Prologo

Este texto está fundamentado en apuntes, y problemas de clase, producto de nuestra experiencia docente universitario en el curso de Programación de Computadoras para ingeniería que se dicta en la Facultad de Ingeniería Química de la Universidad Nacional del Callao.

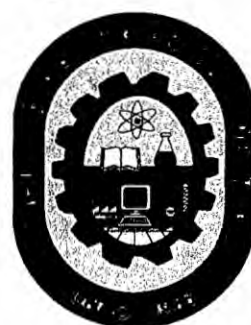
Se busca ofrecer a los estudiantes universitarios de las diversas disciplinas académicas, un enfoque motivado de los principios básicos de la Programación. Es así como una variedad de problemas resueltos aplicados a la ingeniería química que son resueltos con el lenguaje de programación Visual Basic.net, a la vez se presentan sencillez y de tal manera que el alumno logre entenderlos y manejarlos en forma adecuada.

Para cada capítulo se hace una descripción de las principales sentencias que se utilizan y luego los problemas resueltos con el diseño de los formularios y la codificación en los lenguajes de programación Visual Basic.

En el último capítulo se presentan problemas de varios tipos y niveles de dificultad.

Texto:

Problemas en Visual Basic.net



Aplicados a la Ing. Química

*Lic. Salvador Trujillo Pérez
Profesor de la Facultad de Ingeniería
Química
Universidad Nacional del Callao
Setiembre 2009*

*A mi querida esposa Elena
Y a mi querido hijo Omar y Saúl*

Prologo

Este texto está fundamentado en apuntes, y problemas de clase, producto de nuestra experiencia docente universitario en el curso de Programación de Computadoras para ingeniería que se dicta en la Facultad de Ingeniería Química de la Universidad Nacional del Callao.

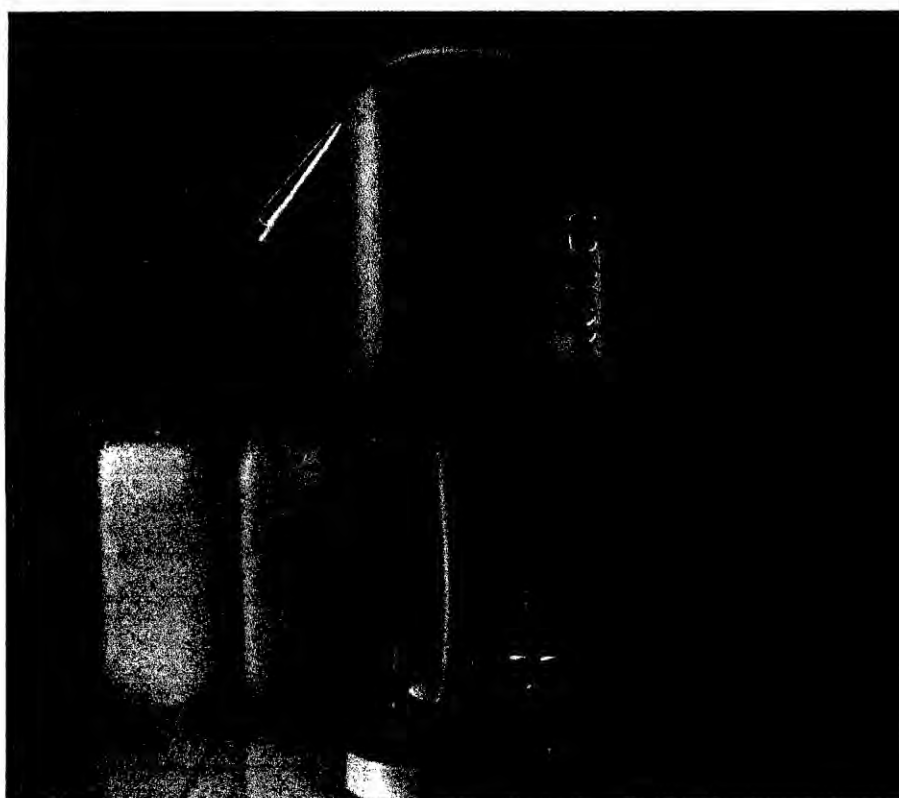
Se busca ofrecer a los estudiantes universitarios de las diversas disciplinas académicas, un enfoque motivado de los principios básicos de la Programación y aplicaciones a problemas químicos que son resueltos con el lenguaje de programación Visual Basic .net, a la vez se presentan sencillez y de tal manera que el alumno logre entenderlos y manejarlos en forma adecuada.

Para cada capítulo se resuelven una variedad de problemas aplicados a la ingeniería química con su respectivo diseño del formulario, y codificación en Visual basic.net.

En el capítulo final se presenta problemas propuestos de varios tipos y niveles de dificultad.

Texto:

Problemas en Visual Basic.net



Aplicados a la Ing. Química

*Lic. Salvador Trujillo Pérez
Profesor de la Facultad de Ingeniería
Química
Universidad Nacional del Callao
Setiembre 2009*

INDICE

1. Capítulo I	1
1.1 Estructura de un programa VB.NET	1
1.1.1 Instalación de Visual Studio.Net	1
1.1.2 Uso de Controles	6
(Aplicación modo Windows)	
Ejemplos Aplicativos	9
2. Capítulo II	16
2.1 Sentencias de Control	16
2.1.1 Estructuras de Decisión	16
a. If...Then	16
Ejemplos Aplicativos	17
b. If...Then...Else	23
Ejemplos Aplicativos	24
c. Select Case	47
Ejemplos Aplicativos	48
2.1.2 Estructuras de Repetición	58
a. Do...Loop	58
Ejemplos Aplicativos	59
b. For...Next	61
Ejemplos Aplicativos	62

3. Capítulo III	75
3.1 Arreglos Unidimensionales	75
Ejemplos Aplicativos	77
4. Capítulo IV	92
4.1 Arreglos Bidimensionales (Matrices)	92
Ejemplos Aplicativos	94
5. Capítulo V	108
5.1 Procedimientos	108
Ejemplos Aplicativos	109
6. Capítulo VI	122
6.1 Funciones	122
6.2 Ejemplos Aplicativos	123
7. Capítulo VII	138
7.1 Problemas propuestos a la Ingeniería química	138
Bibliografía	161

1.- CAPITULO I

1.1.- Estructura de un Programa Visual Basic.net

1.1.1 Instalación de Visual Studio.Net

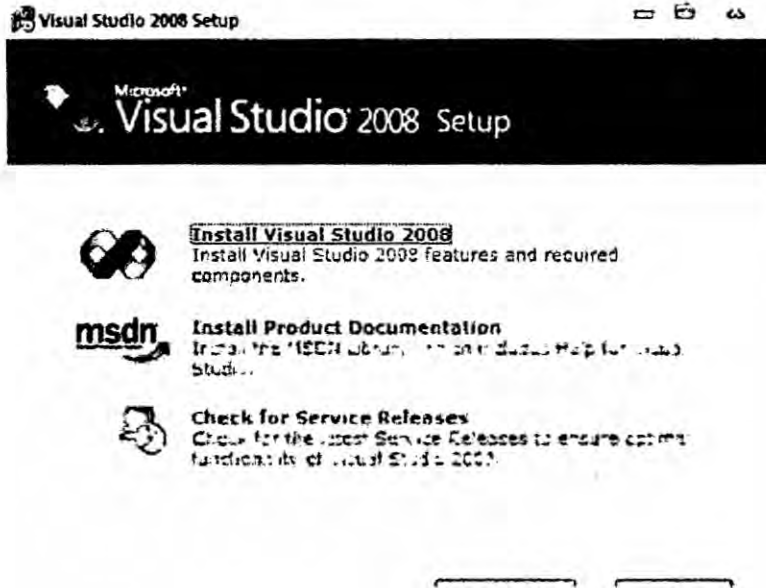
A continuación se muestra la instalación de Visual Studio 2008:

- a).- Después de iniciar sesión en Windows, introduce el disco en la unidad de cd de tu computadora.
- b).- Espera a que aparezca el cuadro de diálogo sobre la pantalla (figura-1):



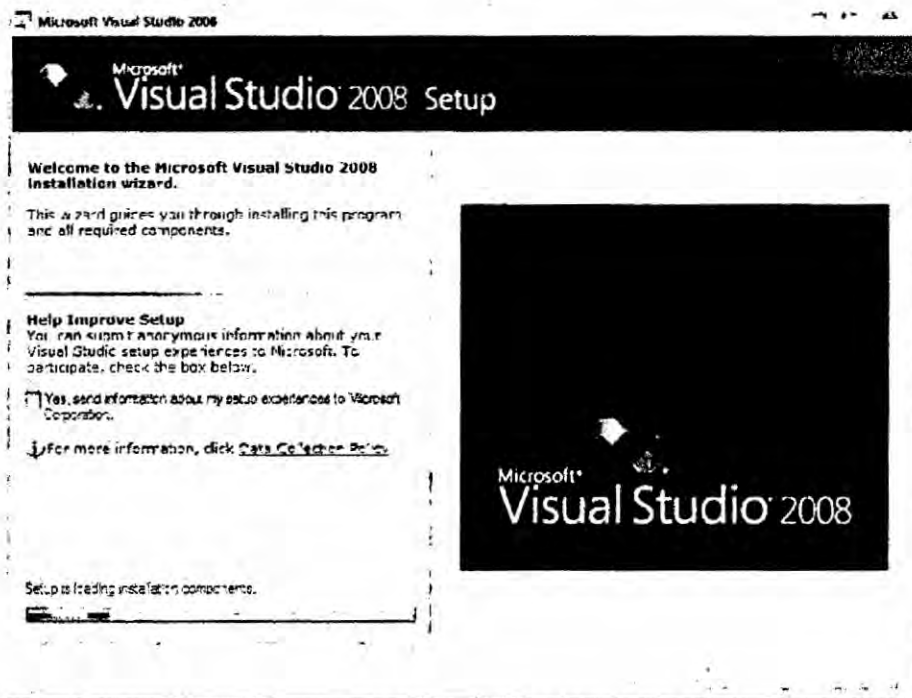
(fig.-1)

c).- Da un click en el ícono que dice "Install Visual Studio 2008" (figura-2)



(fig-2)

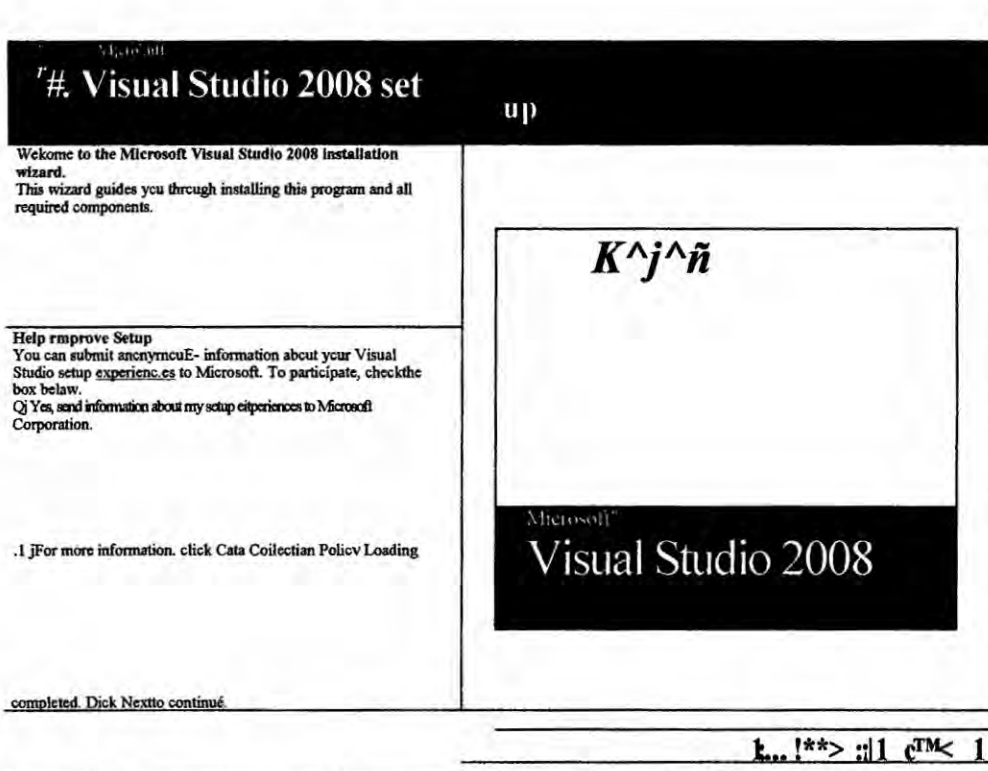
d.) Proceso en la barra verde. (figura-3)



(Fig-3)

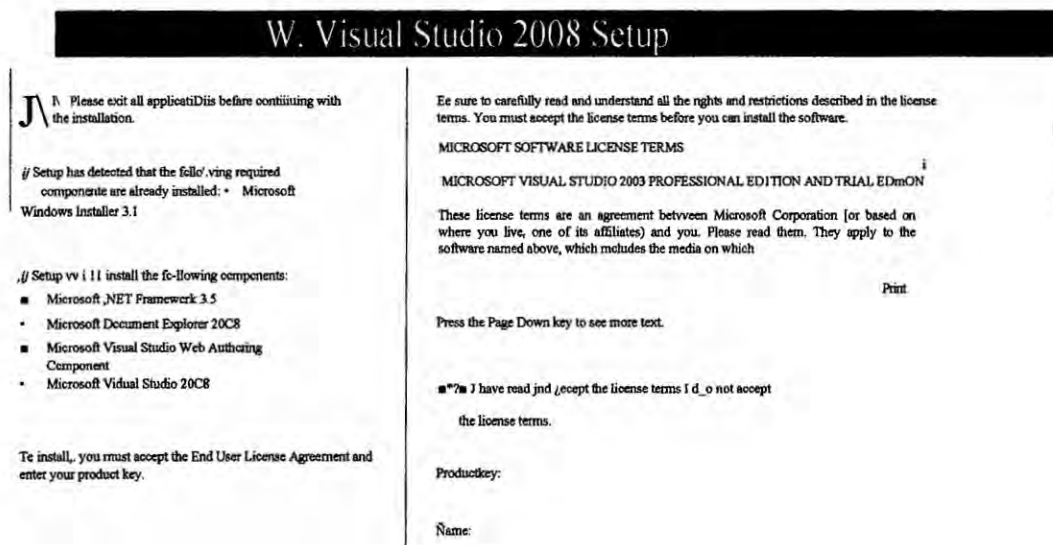
SPP

e).- Espera a que aparezca la siguiente pantalla y da click en el botón de "Next" o "Siguiente" (figura-4)



(Fig-4)

f.) Aceptar en los terminos en que se da el contrato de uso de licencia click en el botón de Next o "Siguiente" (figura-5)



(Fig-5)

Handwritten signature or initials.

g).- Debe aparecer la pantalla que se muestra a continuación; Da un click en el lado izquierdo de la pantalla, en el botón de "Full" y después da un click en el botón de "Install" o "Instalar" (figura-6)

Microsoft Visual Studio 2008 Setup

Select features to install: Default
Installs the recommended features for the product

m
Installs all features for the product

Custom
Select features to include and exclude from the product

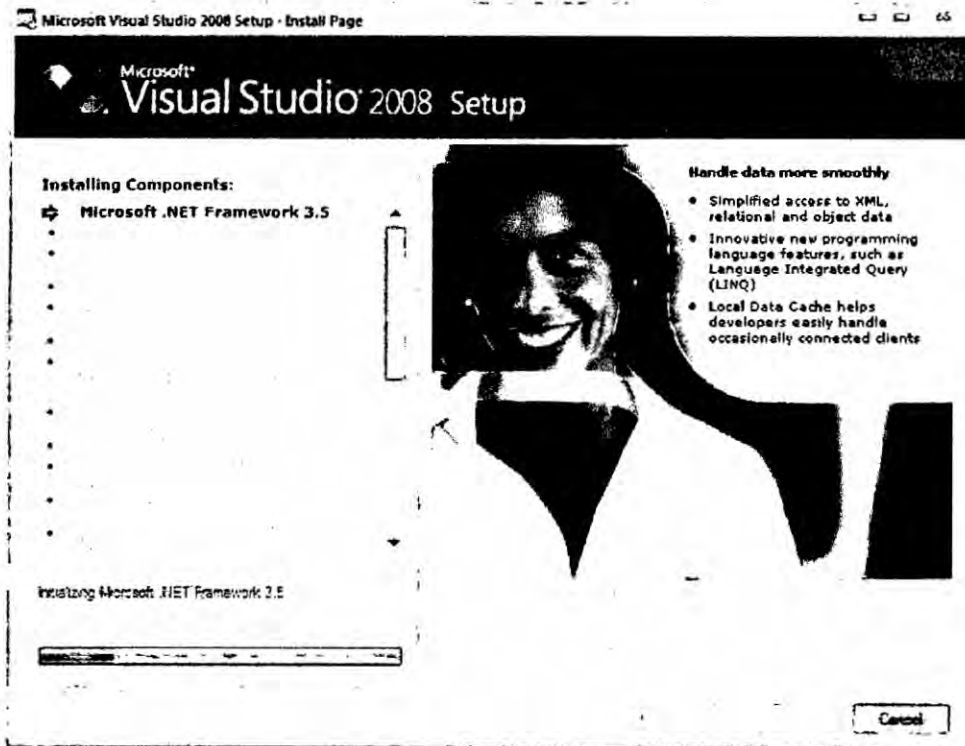
Feature description:
This option installs all of the features available for the product.

Product install path:
E:\Archivos de programa\Microsoft Visual Studio 5.0\

Disk space requirements:

Volume	Disk Size	Available	Required	Remaining
C:	5.8GB	1.3GB	41GB	-
Q:	2.2GB	27.5GB	8.1GB	274 MB

8.- Deberá aparecer la pantalla que se muestra a continuación, espera a que concluya el proceso. (figura-7)



(fig.-7)

SHP

1.1.2 Uso de controles (Aplicación modo Windows)

Una vez instalado el Visual Basic .net , vamos a detallar los pasos que se debe realizar para resolver un problema.

Primero hacer clic en la opción Visual studio 2008 / Archivo / Nuevo proyecto, debe estar seleccionada la opción de Aplicación de Windows Form / y escribir el nombre del Archivo y luego aparece una pantalla como se muestra en la figura-8, debe aparecer a lado izquierdo la Barra de Herramientas y al lado derecho la ventana de propiedades y al centro un cuadro para diseñar el Formulario.

De la barra de herramientas los controles principales son los siguientes:

Label: Este control se utiliza para escribir cadenas de caracteres

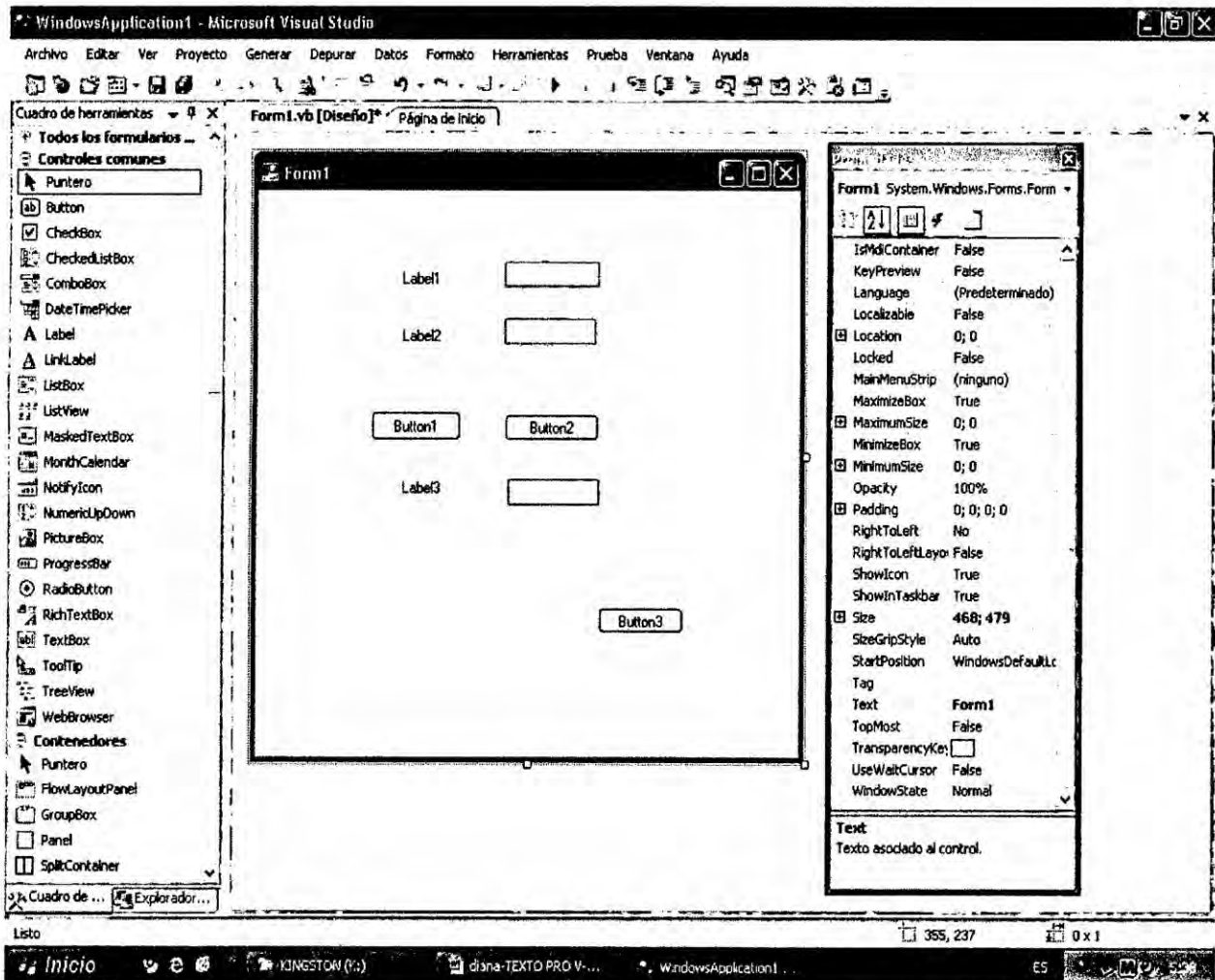
Textbox: Se utiliza para ingresar y/o mostrar información

Button: Se utiliza para ejecutar los procedimientos de Calcular; Borrar, Y Salir

Listbox. Se utiliza para mostrar una lista de datos ingresados.

Con estos controles podemos diseñar el formulario siguiente:

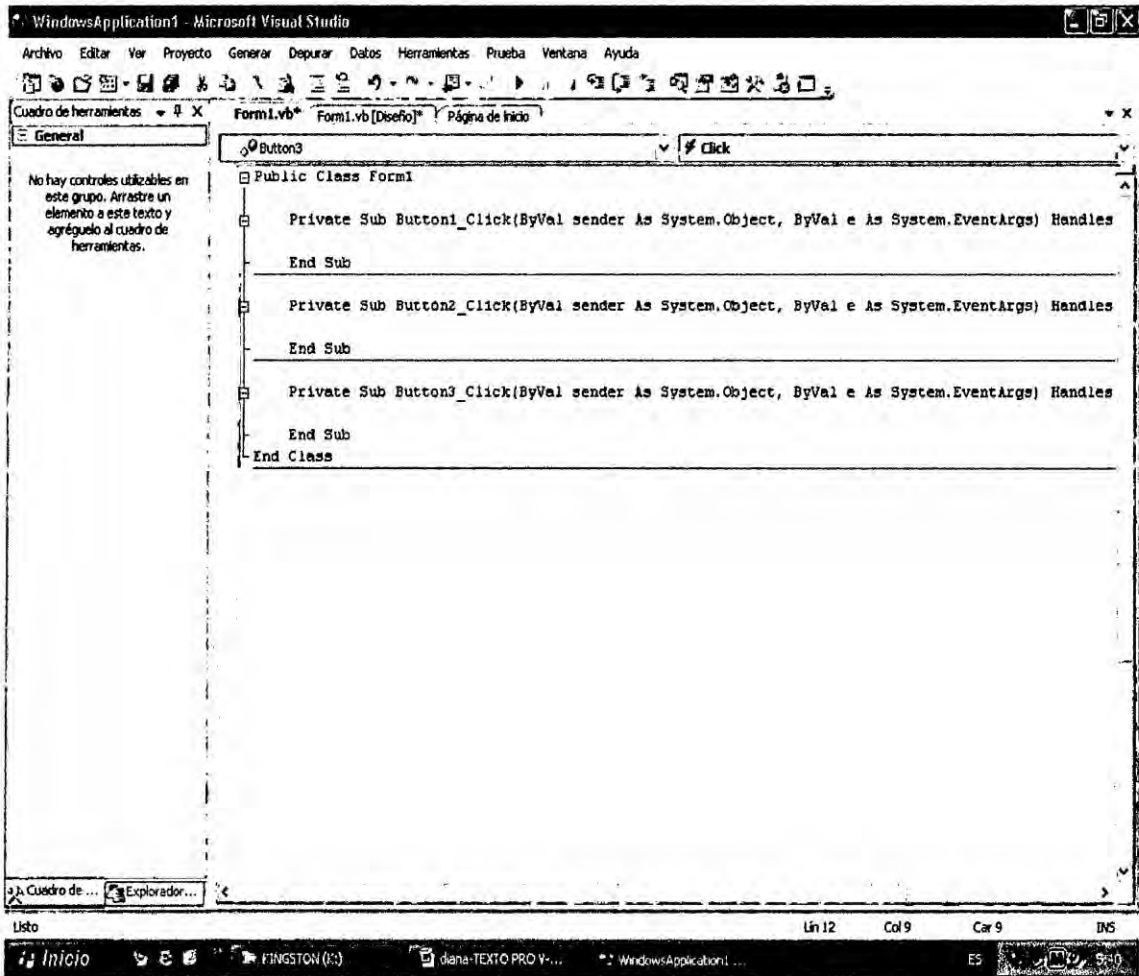




(Fig-8)

Segundo para la codificación en el lenguaje de programación Visual basic.net, dar doble Clic en el Button1, Button, Button3,.. cantidad de botones como se desea programar y el visual creara en forma automática los procedimientos para poder codificar, tal como lo muestra la siguiente figura-9

Signature



(Fig-9)

SHP

EJEMPLOS APLICATIVOS

APLICACIÓN 1

1.- ENUNCIADO DEL PROBLEMA:

Crear un programa en Visual Basic que permita conocer el Trabajo Útil y la eficiencia en una maquina térmica al conocer la temperatura del foco caliente y la temperatura del foco frío.

DATOS:

Sean las variables: TC: Temperatura del foco caliente, variable numérica real

TF: temperatura del foco frío, variable numérica real.

W: trabajo útil, variable numérica real.

EF: eficiencia, variable numérica real.

$$W = T_c - T_f$$

$$EF = \frac{W}{T_c} \times 100\%$$

2.- DISEÑO FORMULARIO-01

The screenshot shows a Windows-style window titled "Termodinámica". It contains a form with the following elements:

- Two input fields for "TEMPERATURA DEL FOCO CALIENTE" and "TEMPERATURA DEL FOCO FRIO".
- Two buttons: "CALCULAR" and "BORRAR".
- Two output fields for "W UTIL" and "EFICIENCIA".
- A "SALIR" button at the bottom.

3.- CODIFICACION

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button1.Click
```

```
    Dim TC, TF, WU, EF As Double
```

```
    TC = TextBox1.Text
```

```
    TF = TextBox2.Text
```

```
    W = TC - TF
```

```
    TextBox3.Text = WU
```

```
    EF = (WU / TC) * 100
```

```
    TextBox4.Text = EF
```

```
End Sub
```

```
End Class
```



APLICACIÓN 2

1.- ENUNCIADO DEL PROBLEMA:

Escribir un programa en Visual Basic, que me halle las variables de la primera Ley de la Termodinámica: Q,W,E ; Teniendo como datos las variables de volumen inicial, volumen final, numero de moles y presión (cte)

DATOS:

$$E = \frac{3}{2} \times N \times R \times (T_f - T_i)$$

$$W = P \times (V_f - V_i)$$

$$Q = E + W$$

$$R = 0.082$$

$$T_f = \frac{(P \times V_f)}{R \times N}$$

$$T_i = \frac{(P \times V_i)}{R \times N}$$

DONDE:

N= Variable numérica tipo real, define el numero de moles

P= Variable numérica tipo real, define la presión

V_i= Variable numérica tipo real, define el volumen inicial

V_f= Variable numérica tipo real, define el volumen final

T_i= Variable numérica tipo real, define el tiempo inicial

T_f= Variable numérica tipo real, define el tiempo final

2.- DISEÑO FORMULARIO-02

Form1

1ERA LEY DE LA TERMODINAMICA

Presion (cte) nro moles

Volumen inicial

Volumen Final

Calcular

1era Ley "Q = E + W"

W =

E =

Q =

Borrar Salir

3.- CODIFICACION

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button3.Click
```

```
Dim w, p, vf, vi, tf, ti, r, n, en, q As Double
```

```
p = Val(TextBox1.Text)
```

```
n = Val(TextBox2.Text)
```

```
r = 0.082
```

```
vi = Val(TextBox3.Text)
```

```
vf = Val(TextBox4.Text)
```

```
ti = (p * vi) / (r * n)
```

```
tf = (p * vf) / (r * n)
```

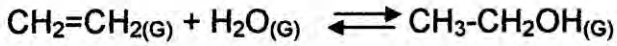
```
w = p * (vf - vi)
TextBox7.Text = Val(w)
en = 1.5 * n * r * (tf - ti)
q = en + w
TextBox5.Text = Val(q)
TextBox6.Text = Val(en)
End Sub
```



APLICACIÓN 3

1.- ENUNCIADO DEL PROBLEMA:

Dado la siguiente reaccion



Se cumple la siguiente ecuacion:

$$\text{LnK}_p = (4926.64 / T) - 14.43$$

Donde:

K_p : Constante de equilibrio de R x n a la temperatura T

$$\text{LnK}_p = r$$

T : temperatura (K)

Hacer un programa, que dado una temperatura, calcule la constante de equilibrio de la reaccion a esa determinada temperatura

2.- DISEÑO FORMULARIO-03

The image shows a graphical user interface window titled "form1". The window contains the following elements:

- Title bar: "form1" with standard window control buttons (minimize, maximize, close).
- Text: "CONSTANTE DE EQUILIBRIO DEL : ETILENO + AGUA = ETANOL"
- Input field: "Temperatura" followed by a text box.
- Buttons: "Calcular" and "Borrar" below the temperature input.
- Output field: "Constante de equilibrio" followed by a text box.
- Button: "Salir" at the bottom right.

5/18

3.- CODIFICACION

Public Class Form1

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

Dim t, r, kp As Double

t = TextBox1.Text

r = 4926.64 / t - 14.34

kp = (2.718281828) ^ r

TextBox2.Text = kp

End Sub

End Class



CAPITULO II

2.1 Sentencias de Control

Las sentencias de control, denominadas también estructuras de control, permiten tomar decisiones y realizar un proceso repetidas veces. Visual Basic.net dispone de las siguientes estructuras:

2.1.1. Estructura de decision

Sentencia If .. Then

Permite tomar una decisión referente al camino a seguir o acción a ejecutar en un proceso basándose en el resultado (verdadero o falso) de una condición. Su sintaxis es:

If condición then

acción1

end if

Donde condición debe ser una expresión numérica, relacional o lógica. Si la condición es verdadera se ejecuta la acción1.

EJEMPLOS APLICATIVOS

APLICACIÓN 4

1.- ENUNCIADO DEL PROBLEMA:

Una industria química produce cantidades de ácido clorhídrico, crear un programa que calcule la mayor cantidad de HCl producida, la menor cantidad, el promedio de las cantidades producidas y la veces que se han producido HCl en la industria, así como se muestra en la siguiente tabla:

Nº DE VECES	CONCENTRACION HCl (m)
3	0.001
4	0.0016
2	0.0024
3	0.018
4	0.03
1	0.032
5	0.05
3	0.1
4	0.15
2	0.2

Tabla-1

2.- DISEÑO FORMULARIO-04

Form1

CANTIDAD DE

ACEPTAR CALCULAR

MAYOR PRODUCCION

MENOR PRODUCCION

NUMERO DE PRODUCCION

PROMEDIO DE PRODUCCION

BORRAR SALIR

3.- CODIFICACION

```
Public S, X, WMAY As Double
```

```
Public WMEN As Double = 99999
```

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button1.Click
```

```
Dim CANT As Integer
```

```
CANT = TextBox1.Text
```

```
ListBox1.Items.Add(CANT)
```

```
If CANT > WMAY Then
```

```
WMAY = CANT
```

```
End If
```

```
If CANT < WMEN Then
```

```
WMEN = CANT
```

End If

S = S + CANT

X = X + 1

TextBox1.Text = ""

TextBox1.Focus()

End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click

Dim PROM As Double

PROM = S / X

TextBox2.Text = WMAY

TextBox3.Text = WMEN

TextBox4.Text = X

TextBox5.Text = PROM

End Sub

End Class



APLICACIÓN 5

1.- ENUNCIADO DEL PROBLEMA:

En una empresa que se dedica a la distribución de sales puras para diferentes industrias y casas, especialmente la sal NaCl obteniéndola a partir de la reacción de HCl con el NaOH, nos dicen lo siguiente: debemos hallar la cantidad de hidróxido de sodio (NaOH) necesario para cierta cantidad requerida de NaCl (cloruro de sodio), sabiendo que existe 50% de porcentaje de pureza de NaOH que se encuentra en un almacén, y que a partir de 1000Kg de NaCl requerido, es solo para uso industrial y menores cantidades solo uso domestico.

Datos:

Si a partir de la reacción: $\text{NaOH} + \text{HCl} \rightleftharpoons \text{NaCl} + \text{H}_2\text{O}$

Supongamos:

m: masa de NaOH del almacén

x: masa de $\text{NaCl}_{\text{puro}}$ requerido por la industria o casa

Pero de esta reacción la cantidad de NaOH y NaCl están en la misma cantidad de moles es decir si queremos 1mol-Kg. de NaCl (75.5kg de NaCl) debemos tener 1mol-Kg. de NaOH (57kg de NaOH)

Ahora si necesitamos "x" de $\text{NaCl}_{\text{puro}}$ entonces necesitamos:

75.5kg de $\text{NaCl}_{\text{puro}}$ \longrightarrow 57kg de $\text{NaOH}_{\text{PURO}}$

"x" de $\text{NaCl}_{\text{puro}}$ \longrightarrow Masa de $\text{NaOH}_{\text{PURO}}$

Masa de $\text{NaOH}_{\text{PURO}} = x * 57/75.5$

Pero "m" (masa de NaOH del almacén) con 50% de pureza

"m " \longrightarrow 100%

Masa de NaOH_{PURO} = $x * 57/75.5$ \longrightarrow 50%

$m = 2 * (x * (57/75.5))$... relación necesaria para el calculo de NaCl puro.

2.- DISEÑO FORMULARIO-05

3.- CODIFICACION

```
Public Class Form1
```

```
    Inherits System.Windows.Forms.Form
```

```
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
```

```
System.EventArgs) Handles Button1.Click
```

```
        Dim x, m As Double
```

```
        x = TextBox1.Text
```

```
        If x > 0 Then
```

```
            m = 2 * 57 * (x / 75.5)
```

```
        End If
```

```
If x >= 1000 Then
    TextBox3.Text = "industrial"
Else
    TextBox3.Text = "domestico"
End If
TextBox2.Text = m
End Sub
End Class
```



b.) Sentencia If... Then Else

Permite tomar una decisión referente al camino a seguir o acción a ejecutar en un proceso basándose en el resultado (verdadero o falso) de una condición.

Su sintaxis es:

If condición then

 acción1

Else

 acción2

End if

Donde condición debe ser una expresión numérica, relacional o lógica. Si la condición es verdadera se ejecuta la **acción1** y si es falsa se ejecutará la **acción2**.

EJEMPLOS APLICATIVOS

APLICACIÓN 6

1.- ENUNCIADO DEL PROBLEMA:

Generar un programa en la cual a través de una ecuación dada, conociendo sus relaciones estequiometrias y las moles de los reactantes podamos calcular:

- Reactivo en exceso (Re)
- Reactivo limitante (RI)
- Moles del reactivo limitante (NI)
- Moles finales en los reactantes producidos por el reactivo limitante (Nf-l)
- Moles de los productos (Np)
- Moles totales al final de la reacción (Nt)

La reacción a hallar es:

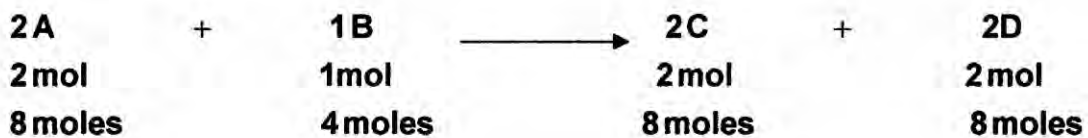


Figura -10

Handwritten signature

2.- DISEÑO FORMULARIO-06

Form1

ESTEQUIOMETRIA

$2A + 1B \dots\dots\dots 2C + 2D$

MOLES

A

B

RESULTADOS

REACTIVO EN EXCESO

REACTIVO LIMITANTE

MOLES DEL REACT. LIMITANTE

MOLES FINALES (REACTANES)

MOLES DE PRODUCTOS

C

D

MOLES TOTALES FINAL

$N_a = \text{moles de A}$

$N_b = \text{moles de B}$

	$N_a = 2 n_b$	$N_a > 2 n_b$	
Reactivo en exceso	No existe	A	B
Reactivo limitante	Ambos	B	A
Moles limitante	Ambos limitan	N_b	N_a
Moles exceso	0	$N_a - 2N_b$	$N_b - \frac{1}{2} \times N_a$
Moles C	N_a	$2 \times n_l$	N_l
Moles D	N_a	$2 \times n_l$	N_a

Tabla -2

3.- CODIFICACION:

Public Class Form1

Inherits System.Windows.Forms.Form

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

Dim nA, nB, nC, nD, nEX, nT As Double

Dim REX, RL As String

Dim ML As Object

MA = TextBox1.Text

MB = TextBox2.Text

If MA = 2 * (MB) Then

REX = "NO EXISTE"

RL = "AMBOS"

ML = "AMBOS LIMITAN"

MEX = 0

MC = MA

MD = MA

Elseif MA > 2 * MB Then

REX = "A"

RL = "B"

ML = MB

MEX = MA - 2 * MB

MC = 2 * ML

MD = 2 * ML

Else

REX = "B"

RL = "A"

ML = MA

MEX = MB - 1 / 2 * (MA)

MC = ML

MD = MA

End If

MT = MC + MD

TextBox3.Text = REX

TextBox4.Text = RL

TextBox5.Text = ML

TextBox6.Text = MEX

TextBox7.Text = MC

TextBox8.Text = MD

TextBox9.Text = MT

End Sub

End Class



APLICACIÓN 7

1.- ENUNCIADO DEL PROBLEMA:

Crear un programa que nos pueda facilitar los cálculos para la titulación del NaOH y HCl ingresando solo sus concentraciones y volúmenes; que nos pueda generar resultados de nuestras experiencias en laboratorio.

Este programa nos debe generar:

- PH (indicar si es mayor que 7, igual a 7 o menor que 7)
- Nos de los valores de moles de nH que se mezclan
- Nos de los valores de moles de nOH que se mezclan
- Nos indique la coloración usando el indicador fenolftaleína
- Qué tipo de solución (ácida, básica o neutra)
- Cuál es el reactivo en exceso
- POH (indicar si es mayor que 7, igual a 7 o menor que 7)

DATOS:

$$Ph + POh = 14$$

$$\text{HCl} \begin{cases} x = \text{concentración} \\ Vx = \text{Volumen} \end{cases}$$

$$\text{NaOH} \begin{cases} y = \text{concentración} \\ X = \text{volumen} \end{cases}$$

Rango de pH	Color de la fenolftaleína	Tipo de solución	Reactivo en exceso
pH > 7	Rojo Grosella	Básica	NaOH
pH = 7	Rosado Claro	Neutra	No existe
pH < 7	Transparente	Ácida	HCl

Tabla -3

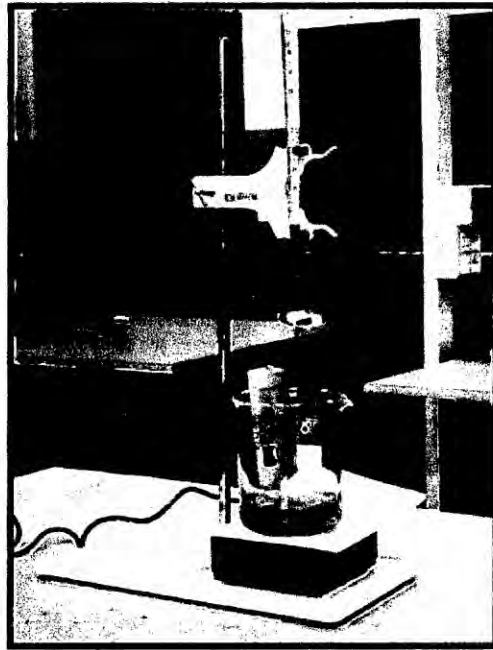


Figura -11

2.- DISEÑO FORMULARIO-07

Form1
[-] [x]

ACIDOS - BASE

$\text{NaOH} + \text{HCl} \longrightarrow \text{NaCl} + \text{H}_2\text{O}$

	CONCENTRACION (MOL/L)	VOLUMEN (L)
HCl	<input style="width: 80%;" type="text"/>	<input style="width: 80%;" type="text"/>
NaOH	<input style="width: 80%;" type="text"/>	<input style="width: 80%;" type="text"/>

RESULTADOS

PH	<input style="width: 80%;" type="text"/>
MOLAS DE HCl (M)	<input style="width: 80%;" type="text"/>
MOLAS DE NaOH (M)	<input style="width: 80%;" type="text"/>
MOLAS (GEN. PL. F. M.)	<input style="width: 80%;" type="text"/>
TIPO DE REACCION	<input style="width: 80%;" type="text"/>
VOL. HCl (L)	<input style="width: 80%;" type="text"/>
RELACION EQUIVALENCIA	<input style="width: 80%;" type="text"/>
PH	<input style="width: 80%;" type="text"/>

SPB

3.- CODIFICACION

Public Class Form1

Inherits System.Windows.Forms.Form

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As

System.EventArgs) Handles Button1.Click

Dim nH, nOH, VT, X, Y, Vx, Vy As Double

Dim C, TS As String

Dim PH, REX As Object

X = TextBox1.Text

Y = TextBox2.Text

Vx = TextBox3.Text

Vy = TextBox4.Text

nH = X * Vx

nOH = Y * Vy

If nH = nOH Then

PH = 7

C = "ROSADO CLARO"

TS = "SOL. NEUTRA"

REX = " NO EXISTE"

Elseif nH < nOH Then

PH = " MAYOR QUE 7"

C = " ROJO GROSELLA"

TS = "SOL. BASICA"

REX = " NaOH "

Else

PH = " MENOR QUE 7"

C = " TRANSPARENTE"

TS = " SOL. ACIDA"

REX = " HCl "

End If

$VT = Vx + Vy$

TextBox5.Text = PH

TextBox6.Text = nH

TextBox7.Text = nOH

TextBox8.Text = C

TextBox9.Text = TS

TextBox10.Text = VT

TextBox11.Text = REX

End Sub

End Class



APLICACIÓN 8

1.- ENUNCIADO DEL PROBLEMA:

En un cierto proceso industrial se requiere oxígeno y nitrógeno, y se necesita realizar un programa especial para calcular el precio de estos gases con respecto a la política de la distribuidora. La distribuidora de estas sustancias usa la política siguiente cada litro de oxígeno cuesta 65 soles, si se requiere más de 50 litros de oxígeno el costo será 10% menos que el total requerido, si se desea más de 100 litros será 20% menos del total requerido. Si se requiere nitrógeno el litro cuesta 40soles, si se desea más de 60litros el costo será 15% menos del total requerido de nitrógeno.

Si el volumen de oxígeno entregado para un proceso industrial es mayor que el volumen de nitrógeno el proceso será considerado deficiente de lo contrario es eficiente.

DATOS:

	Volumen	Costo (soles)
Oxígeno	$V_o < 50$	$C_o = S/.65 \times V_o$
	$50 \leq V_o < 100$	$C_o = s/. 65 \times V_o - 0.1 \times s/.65 \times V_o$
	$V_o \geq 100$	$C_o = s/. 65 \times V_o - 0.2 \times s/.65 \times V_o$
Nitrogeno	$V_n < 60$	$C_n = s/. 40 \times V_n$
	$V_n \geq 60$	$C_n = s/. 40 \times V_n - 0.15 \times s/. 40 \times V_n$

Tabla - 4

Por lo tanto:

Tipo de proceso: $V_o > V_n \rightarrow$ deficiente

$V_o < V_n \rightarrow$ eficiente

vo: volumen de oxígeno

vn: volumen de nitrógeno

co: costo del oxígeno

cn: costo del nitrógeno.

2.- DISEÑO FORMULARIO-08

The image shows a screenshot of a Windows application window titled "Form1". The window contains a form with several input fields and buttons. The form is organized into two main sections. The top section, which has a dotted background, contains two text boxes labeled "Volumen de oxígeno" and "Volumen de nitrógeno". Below these are two buttons labeled "Calcular" and "Borrar". The bottom section contains four text boxes labeled "Costo del oxígeno", "Costo del nitrógeno", "Costo total", and "Tipo de proceso". A "Salir" button is located at the bottom right of the form.

3.- CODIFICACION

```
Public Class Form1
```

```
    Inherits System.Windows.Forms.Form
```

```
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
```

```
System.EventArgs) Handles Button1.Click
```

```
        Dim vo, vn, co, cn As Double
```

```
        vo = TextBox1.Text
```

```
        vn = TextBox2.Text
```

```
        If vo > vn Then
```

```
            TextBox6.Text = "deficiente"
```

```
        Else
```



```
    TextBox6.Text = "eficiente"
```

```
End If
```

```
If vo < 50 Then
```

```
    co = 65 * vo
```

```
Elseif vo >= 50 And vo < 100 Then
```

```
    co = 65 * vo - 0.1 * 65 * vo
```

```
Elseif vo >= 100 Then
```

```
    co = 65 * vo - 0.2 * 65 * vo
```

```
End If
```

```
If vn < 60 Then
```

```
    cn = vn * 40
```

```
Elseif vn >= 60 Then
```

```
    cn = vn * 40 - 0.15 * 40 * vn
```

```
End If
```

```
TextBox3.Text = co
```

```
TextBox4.Text = cn
```

```
TextBox5.Text = co + cn
```

```
End Sub
```

```
End Sub
```

```
End Class
```



APLICACIÓN 9

1.- ENUNCIADO DEL PROBLEMA:

Diseñar un programa que nos determine la relación de color absorbido y color percibido según el espectro de emisión (luz visible) ,a partir de su longitud de onda cuyo rango varia desde 380nm hasta 770nm, ver siguiente grafica

LONGITUD DE ONDA(nm)	COLOR ABSORBIDO	COLOR PERCIBIDO
380-459	VIOLETA-AZUL	VERDE-AMARILLO
459-488	AZUL	AMARILLO
488-515	VERDE-AZUL	ANARANJADO-ROJO
515-571	VERDE-AMARILLO	VIOLETA
571-582	AMARILLO	AZUL
582-596	AMARILLO-AZUL	VERDE-AZUL
596-770	ANARANJADO-ROJO	VERDE

Tabla -5

2.- DISEÑO FORMULARIO-09

The screenshot shows a Windows form titled "Form1" with the following elements:

- A text label "LONGITUD DE ONDA" followed by a text box containing the value "525".
- Two buttons: "CALCULAR" and "BORRAR".
- A text label "COLOR_ABSORBIDO" followed by a text box containing the value "VERDE_AMARILLO".
- A text label "COLOR_PERCIBIDO" followed by a text box containing the value "VIOLETA".
- A button labeled "FIN" at the bottom center.

3.- CODIFICACION

Public Class Form1

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

Dim L As Integer

Dim ABSOR, PERCIB As String

L = TextBox1.Text

If L > 380 And L <= 459 Then

ABSOR = "VIOLETA_AZUL"

PERCIB = "VERDE_AMARILLO"

Elseif L > 459 And L <= 488 Then

ABSOR = "AZUL"

PERCIB = "AMARILLO"

Elseif L > 488 And L <= 515 Then

ABSOR = "VERDE_AZUL"

PERCIB = "ANARANJADO_ROJO"

Elseif L > 515 And L <= 571 Then

ABSOR = "VERDE_AMARILLO"

PERCIB = "VIOLETA"

Elseif L > 571 And L <= 582 Then

ABSOR = "AMARILLO"

PERCIB = "AZUL"

Elseif L > 582 And L <= 596 Then

ABSOR = "AMARILLO_AZUL"

```
PERCIB = "VERDE_AZUL"  
Elseif L > 596 And L <= 770 Then  
    ABSOR = "ANARANJADO_ROJO"  
    PERCIB = "VERDE"  
Else  
    ABSOR = "ERROR"  
    PERCIB = "ERROR"  
End If  
TextBox2.Text = ABSOR  
TextBox3.Text = PERCIB  
End Sub  
End Class
```



APLICACION 10

1.- ENUNCIADO DEL PROBLEMA:

Desarrollar un programa en visual basic, que al ingresar número de moles, calor a presión constante, temperatura inicial, temperatura final, permita determinar el calor de reacción (entalpía) y el tipo de reacción (endotérmica, exotérmica o se trata de un elemento en su estado halo trópico).

DATOS

Estas son las variables a utilizar:

NM = numero de moles

CP = calor a presión Constante

T1 = temperatura inicial

T2 = temperatura final

H = entalpía

entalpia

$$H = NM * CP * (T2 - T1)$$

ENTALPIA	TIPO DE REACCION
> 0	Endotermica
= 0	Elemento
< 0	exotermica

Tabla -6

2.- DISEÑO FORMULARIO-10

ENTALPIAS DE REACCION

NRO DE MOLES	<input type="text"/>	CALCULAR
CP	<input type="text"/>	BORRAR
TEMPERATURA INICIAL	<input type="text"/>	
TEMPERATURA FINAL	<input type="text"/>	
ACEPTAR		
ENTALPIA	<input type="text"/>	
TIPO DE REACCION	<input type="text"/>	SALIR

3.- CODIFICACION

Public Class Form1

Inherits System.Windows.Forms.Form

Public NM, CP, T1, T2, H As Double

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As

System.EventArgs) Handles Button1.Click

NM = Val(TextBox1.Text)

CP = Val(TextBox2.Text)

T1 = Val(TextBox3.Text)

T2 = Val(TextBox4.Text)

H = NM * CP * (T2 - T1)

TextBox1.Text = ""

```
TextBox2.Text = ""
```

```
TextBox3.Text = ""
```

```
TextBox4.Text = ""
```

```
TextBox1.Focus()
```

```
End Sub
```

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
```

```
System.EventArgs) Handles Button2.Click
```

```
    If H > 0 Then
```

```
        TextBox6.Text = "ENDOTERMICA"
```

```
    ElseIf H = 0 Then
```

```
        TextBox6.Text = "ELEMENTO"
```

```
    Else : TextBox6.Text = "EXOTERMICA"
```

```
    End If
```

```
    TextBox5.Text = Val(H)
```

```
End Sub
```

```
End Class
```

APLICACIÓN 11

1.- ENUNCIADO DEL PROBLEMA:

Según el número de electrones, proporcione qué elemento y símbolo tiene.

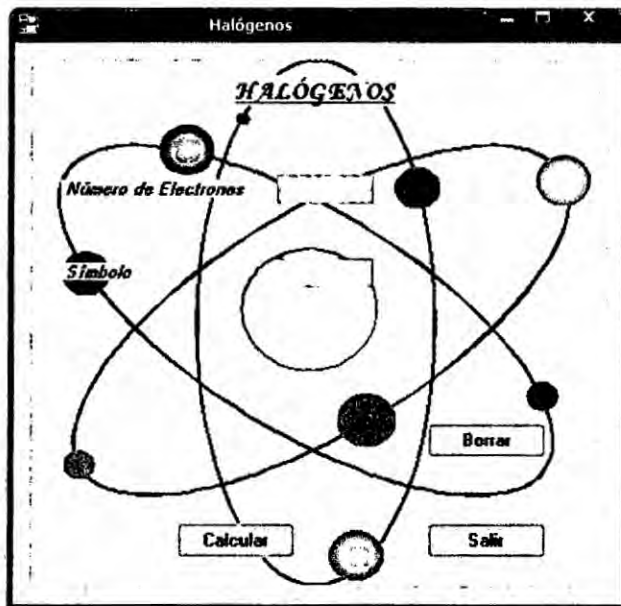
Datos:

- ✓ Halógenos $ns^2 np^5$ n: nivel
- ✓ Fluor : F # electrones = 9
- ✓ Cloro: Cl # electrones = 17
- ✓ Bromo: Br # electrones = 35
- ✓ Yodo: I # electrones = 53
- ✓ Astató: As # electrones = 85

Elemento	# Electrones
Fluor	9
Cloro	17
Bromo	35
Yodo	53
Astató	85

Tabla -7

2.- DISEÑO FORMULARIO-11



3.- CODIFICACION

Public Class Form1

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click

Dim num As Integer

num = TextBox1.Text

Select Case num

Case 9

TextBox2.Text = "F"

Case 27

TextBox2.Text = "Cl"

Case 35

TextBox2.Text = "Br"

Case 53

```
    TextBox2.Text = "I"
```

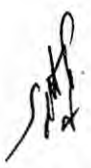
```
Case 127
```

```
    TextBox2.Text = "As"
```

```
End Select
```

```
End Sub
```

```
End Class
```



APLICACIÓN 12

1.- ENUNCIADO DEL PROBLEMA:

Identifica por sus valores de pH si las disoluciones son ácidas o básicas, además a qué color cambia el indicador (papel tornasol).

Datos:

	TIPO DE SOLUCION	RANGO DE pH	COLOR DE PAPEL TORNASOL
pH de 0 a 6	Ácida	$0 < \text{pH} < 6$	Rojo
pH = 7	Neutra	pH = 7	No se altera
pH de 8 a 14	Básica	$8 < \text{pH} < 14$	azul

Tabla - 8

✓ Papel Tornasol:

Rojo → Ácido

No se Altera → Neutro

Azul → Básico

$$\text{pH} = -\log [\text{H}_3\text{O}^+]$$

pH	Disoluciones Ácidas y Básicas
2	Jugo de Limón
5	Agua de Lluvia
7	Agua Pura
9	Levadura
14	Hidróxido de Sodio

2.- DISEÑO FORMULARIO-12

The screenshot shows a graphical user interface for a pH measurement form. The window title is 'pH'. On the left side, there are three text input fields labeled 'Nº pH', 'Clasificación', and 'Color en Papel Tornasol'. On the right side, there is a circular color chart with segments labeled 0 through 10, and the text 'pH 1-10' and 'Clasificar' below it. At the bottom of the window, there are three buttons: 'Borrar', 'Salir', and 'Clasificar'.

3.- CODIFICACION

Public Class Form1

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

Dim pH As Integer

Dim col As String

pH = TextBox1.Text

col = TextBox3.Text

Select Case pH

Case 0, 1, 2, 3, 4, 5, 6

TextBox2.Text = "Ácida"

TextBox3.Text = "Rojo"

Case 7

TextBox2.Text = "Neutra"

TextBox3.Text = "No cambia"

Case 8, 9, 10, 11, 12, 13, 14

TextBox2.Text = "Básica"

TextBox3.Text = "Azul"

Case Else

TextBox2.Text = "ERROR"

TextBox3.Text = "ERROR"

End Select

End Sub

End Class



c.-) Sentencia SELECT

Esta expresión permite ejecutar una de varias acciones en función del valor de una expresión. Es una alternativa a If... Then ... Elself cuando lo que se necesita es comprobar es la misma expresión con diferentes valores. Su sintaxis es:

Select case expresión

```
case lista1
    sentencias
case lista2
    sentencias
case else
    sentencias n
```

end select

Donde expresión es una expresión numérica o alfanumérica, y lista1 y lista2.... representan una lista que puede tener cualquiera de las formas siguientes:

expresión[,expresión]...

expresión to expresión

Is operador-de-relación expresión

combinación de las anteriores separadas por comas

Ejemplos Aplicativos

APLICACIÓN 13

1.- ENUNCIADO DEL PROBLEMA:

Diseñe un programa en el que se ingresan la cantidad de carbonos presentes en un hidrocarburo, y nos permita hallar el nombre de éste; su fórmula molecular y el porcentaje en peso del carbono y el hidrogeno en cada caso, así como se muestra en la siguiente tabla:

NOMBRES	Nº DE CARBONOS
Metano	1
Eteno	2
Propino	3
Buteno	4
Pentano	5
Hexino	6
Hepteno	7
Octino	8
Noneno	9
Decano	10

Tabla - 9

$$X = (2 * C) + 2 \quad \text{Alcano}$$

$$Y = 2 * C \quad \text{Alquenos}$$

$$Z = (2 * c) - 2 \quad \text{Alquino}$$

$$C = \text{Numero de carbonos}$$

2.- DISEÑO FORMULARIO-13

Form1

N. DE CARBONOS

NOM. ALCANO C H %C %H

NOM. ALQUENO C H %C %H

NOM. ALQUINO C H %C %H

3.- CODIFICACION

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button1.Click
```

```
Dim C, X, Y, Z As Integer
```

```
Dim MTA, MTB, MTC, PCA, PCB, PCC, PHA, PHB, PHC As Double
```

```
Dim PREF, NOM_ALCANO, NOM_ALQUENO, NOM_ALQUINO As String
```

```
C = TextBox1.Text
```

```
Select Case C
```

```
Case 1
```

```
PREF = "META"
```

```
Case 2
```

```
PREF = "ET"
```

```
Case 3
```

```
PREF = "PROP"
```

```
Case 4
```


PREF = "BUT"

Case 5

PREF = "PENT"

Case 6

PREF = "HEX"

Case 7

PREF = "HEPT"

Case 8

PREF = "OCT"

Case 9

PREF = "NON"

Case 10

PREF = "DEC"

End Select

NOM_ALCANO = PREF & "ANO"

NOM_ALQUENO = PREF & "ENO"

NOM_ALQUINO = PREF & "INO"

$X = (2 * C) + 2$

$Y = 2 * C$

$Z = (2 * C) - 2$

TextBox2.Text = NOM_ALCANO

TextBox3.Text = NOM_ALQUENO

TextBox4.Text = NOM_ALQUINO

Label12.Text = C

Label11.Text = X

```
Label14.Text = C
Label13.Text = Y
Label16.Text = C
Label15.Text = Z
MTA = (C * 12) + X
MTB = (C * 12) + Y
MTC = (C * 12) + Z
PCA = (C * 1200) / MTA
PHA = (X * 100) / MTA
PCB = (C * 1200) / MTB
PHB = (Y * 100) / MTB
PCC = (C * 1200) / MTC
PHC = (Z * 100) / MTC
TextBox5.Text = PCA
TextBox6.Text = PHA
TextBox7.Text = PCB
TextBox8.Text = PHB
TextBox9.Text = PCC
TextBox10.Text = PHC

End Sub
```

APLICACIÓN 14

1.- ENUNCIADO DEL PROBLEMA:

La distribución de los pares electrones (NRO) alrededor de un átomo central de una molécula está entre los siguientes valores 2, 3, 5 y 6 hacer un programa que devuelva las siguientes clasificaciones para cada par electrónico:

- tipo de hibridación del átomo central
- geometría eléctrica del átomo central
- ejemplo correspondiente del dicho tipo

NRO	HIDRIDACION	GEOMETRIA MOLECULAR	EJEMPLO
2	sp	Lineal	BeCl ₂
3	sp ²	Trigonal plana	BF ₃
4	sp ³	Tetraédrica	CH ₄
5	sp ³ d	Piramidal trigonal	PCl ₅
6	Sp ³ d ²	octaédrica	SF ₆

Tabla - 10

2.- DISEÑO FORMULARIO-14

Form1

NRO

TIPO DE HIBRID

GEOM ELECTRIC

EJEMPLO

3.- CODIFICACION

Public Class Form1

Sub NOMBRAR_HIBRID_GEOM_EJM(ByVal NRO As Integer, ByRef
HIBRID As String, ByRef GEOM As String, ByRef EJM As String)

Select Case NRO

Case 2 : HIBRID = "sp"

GEOM = "LINEAL"

EJM = "BeCl₂"

Case 3 : HIBRID = "sp²"

GEOM = "TRIGONAL PLANA"

EJM = "BF₃"

Case 4 : HIBRID = "sp³"

GEOM = "TETRAEDRICA"

EJM = "CH₄"

Case 5 : HIBRID = "sp³d"

GEOM = "BIPIRAMIDAL"

EJM = "PCl₅"

Case 6 : HIBRID = "sp³d²"

GEOM = "OCTAEDRICA"

EJM = "SF₆"

Case Else

HIBRID = "ERROR"

GEOM = "ERROR"

EJM = "ERROR"

End Select

End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

Dim NRO As Integer

Dim HIBRID, GEOM, EJM As String

NRO = TextBox1.Text

NOMBRAR_HIBRID_GEOM_EJM(NRO, HIBRID, GEOM, EJM)

TextBox2.Text = HIBRID

TextBox3.Text = GEOM

TextBox4.Text = EJM

End Sub

End Class



APLICACIÓN 15

1.- ENUNCIADO DEL PROBLEMA:

En la torre de destilación del petróleo, las temperaturas en la columna de fraccionamiento se regula de forma que esté mas caliente en el fondo y más fría en la parte superior. Como el punto de ebullición aumenta a medida que aumenta el peso molecular, los hidrocarburos mas pesados se condensan cerca del fondo de la columna y los hidrocarburos más ligeros se condensan cerca de la parte superior, las tuberías que están conectadas a diferentes alturas permiten drenar continuamente cada fracción.

Se desea hacer un programa que permita conocer el nombre del producto obtenido a determinada temperatura en un punto de la torre de destilación, asi como se muestra en la siguiente tabla:

NOMBRE DEL PRODUCTO	INTERVALO DE TEMPERATURA DE EBULLICIÓN
Fracción gaseosa o gas natural	< 40°C
Éter de petróleo	40-70°C
Ligroina o nafta ligera	70-100°C
Gasolina/ nafta	100-200°C
Kerosene o Diesel -1 (D-1)	200-280°C
Gasoil o Diesel -2 (D-2)	280-350°C
Fuel-oil o aceite ligero	350-400°C
Fuel-oil o aceite lubricante pesado	400-500°C
Residuo o alquitrán de petróleo	> 500°C

Tabla - 11

T= variable numérica tipo real, define temperatura.

P= variable tipo string, almacena nombre del producto.

2.- DISEÑO FORMULARIO-15



3.- CODIFICACION

```
Dim T As Double
```

```
Dim P As String
```

```
T = TextBox1.Text
```

```
Select Case T
```

```
Case Is <= 40
```

```
    P = "Fracción gaseosa o gas natural"
```

```
Case Is <= 70
```

```
    P = "Éter de petróleo"
```

```
Case Is <= 100
```

```
    P = "Ligroina o nafta ligera"
```

```
Case Is <= 200
```



P = "Gasolina/ nafta"

Case Is <= 280

P = "Kerosene o Diesel -1 (D-1)"

Case Is <= 350

P = "Gasoil o Diesel -2 (D-2)"

Case Is <= 400

P = "Fuel-oil o aceite ligero"

Case Is <= 500

P = "Fuel-oil o aceite lubricante pesado"

Case Else

P = "Residuo o alquitrán de petróleo"

End Select

TextBox2.Text = P

End Sub

2.1.2 Estructura de Repetición

a.) Do Loop

Un Loop(bucle) repite la ejecución de un conjunto de sentencias mientras una condición dada sea cierta, o basta que una condición dada sea cierta. La condición puede ser verificada antes o después de ejecutarse el conjunto de sentencias:

Formato1

Do

[{While|Until}condición]

[sentencias]

[Exit do]

[sentencias]

Loop

Formato2

Do

[sentencias]

[Exit do]

[sentencias]

Loop[{While|Until}condición]

Donde condición es cualquier expresión que se evalúe a True o a False.

EJEMPLOS APLICATIVOS

APLICACIÓN 16

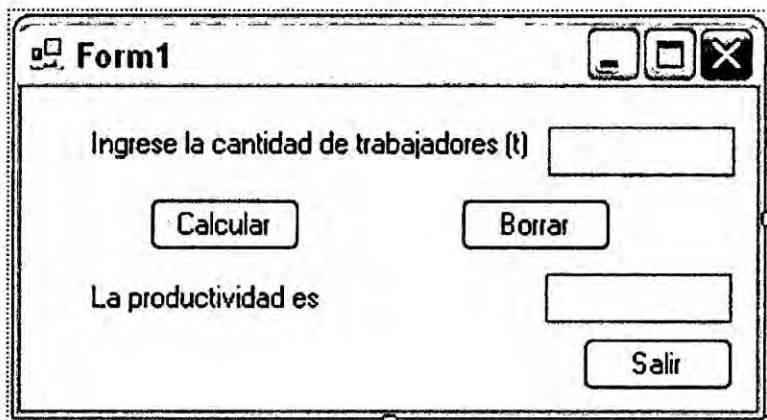
1.- ENUNCIADO DEL PROBLEMA:

En un industria la productividad (P) y la cantidad de trabajadores t, se relaciona en la siguiente ecuación

$$P(t) = \sum_{i=1}^n (t^2 + \log t)$$

Hacer un programa que nos de la productividad para cualquier valor de "t"

2.- DISEÑO FORMULARIO-16



3.- CODIFICACION

```
Public Class Form1
```

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button1.Click
```

```
Dim s, i, t As Double
```

```
i = 1
```

```
s = 0
```

```
t = TextBox1.Text
```

```
While i <= t
```

```
    s = s + (i ^ 2 + Log(10) )
```

```
    i = i + 1
```

End While

TextBox2.Text = s

End Sub

End Class



b.-) Sentencias: For.. Next

La sentencia for da lugar a un bucle que permite ejecutar un conjunto de sentencias cierto número de veces. Su sintaxis es:

for variable = expresion1 **to** expresion2[**Step** expresion3]

[sentencias]

[Exit for]

[sentencias]

Next [variable[,variable...]]

Cuando se ejecuta una sentencia For en la que el valor de la expresión3 es positivo o no se ha especificado, primero se ha asigna el valor de la expresión1 a la variable y a continuación se comprueba si la variable es mayor que la expresión 2, en cuyo caso se salta el cuerpo del bucle y se continúa en la línea que éste a continuación de la sentencia Next. En otro caso, se ejecutan las líneas de programa que haya entre la sentencia For y la sentencia Next. Por último, la variable, se incrementa en el valor de la expresión3, o en 1 si Step no se especifica, volviéndose a efectuar la comparación entre la variable y la expresión2, y así sucesivamente. La sentencia exit for permite salir del bucle for... next antes de que este finalice.

Ejemplos Aplicativos

APLICACION 17

1.- ENUNCIADO DEL PROBLEMA:

Diseñe un programa que deduzca la formula empírica de un compuesto a partir de las masas halladas y de los pesos atómicos de cada elemento químico presente en dicho compuesto, asi como se muestra en la siguiente tabla:

	Masa hallada	Peso atómico
Elemento 1	Masa 1,1	Masa 1,2
Elemento 2	Masa 2,1	Masa 2,2
Elemento 3	Masa 3,1	Masa 3,2

Tabla - 12

Module Module1

Sub Main()

Dim q(9), mtotal, porc, masa(9, 1), dec(9), ind(20), m, p, wmen, nmen As

Double

Dim f, c, n, i, r(9), rmay, indice As Integer

wmen = 100

nmen = 1

Console.WriteLine("Escriba el número de elementos presentes")

n = Console.ReadLine

For f = 0 To n - 1

For c = 0 To 1

masa(f, c) = Console.ReadLine

```
Console.WriteLine("fila==>" & f & "columna==>" & c)
```

```
mtotal = mtotal + masa(f, 0)
```

```
Next
```

```
Next
```

```
For f = 0 To n - 1
```

```
porc = masa(f, 0) / mtotal * 100 * 2
```

```
Console.WriteLine("porcentaje de" & f & "es==>" & porc & "%")
```

```
Next
```

```
For f = 0 To n - 1
```

```
q(f) = masa(f, 0) / masa(f, 1)
```

```
If q(f) < wmen Then
```

```
wmen = q(f)
```

```
End If
```

```
Next
```

```
For f = 0 To n - 1
```

```
dec(f) = q(f) / wmen
```

```
For i = 0 To 20
```

```
ind(i) = dec(f) * (i + 1)
```

```
m = ind(i) - Int(ind(i))
```

```
If m > 0.5 Then
```

```
p = 1 - m
```

```
Else
```

```
p = m
```

```
End If
```

```
If p < nmen Then
```

```
nmen = p
r(f) = i
End If
Next
If r(f) > rmay Then
    rmay = r(f)
End If
nmen = 1
Next
For f = 0 To n - 1
    indice = Int((q(f) / wmen) * (rmay + 1) + 0.3)
    Console.WriteLine("indice" & f & "indice" & rmay)
Next
Console.Read()
End Sub
End Module
```



APLICACIÓN 18

1.- ENUNCIADO DEL PROBLEMA:

En el laboratorio de una industria química que vende 8 tipos de ácidos, llenados en botellas de 700mL, se quiere saber cuantas botellas de cada acido hay, sabiendo además que en cada botella no se pone el nombre del acido, sino la concentración. Para lo cual se le pide a un practicante de la UNAC que haga un inventario de los ácidos.

Como este practicante llevo el curso de programación de computadoras en la universidad, realiza un programa en el cual ingresa las concentraciones de todos los ácidos cuantas veces encuentra, después ingresando la concentración de cualquier acido puede saber el numero de botellas que hay de el algún acido especifico, así como se muestra en la siguiente tabla:

ACIDOS	CONCENTRACIONES
AC. CLORIDRICO	8.2
AC. SULFURICO	4.2
AC. NITRICO	6.8
AC. ACETICO	1.4
AC. CARBONICO	5.6
AC. BROMIDRICO	4.6
AC. OXALICO	8.8
AC. FOSFORICO	9.2

Tabla - 13

El programa funciona ingresando la concentración de los ácidos dados en la tabla, si queremos conocer cuantas botellas de algún ácido, bastara en escribir su concentración en el label2(ACIDO ESPECIFICO).

2.- DISEÑO FORMULARIO-17

The screenshot shows a Windows form titled "Form1" with the following elements:

- A text box labeled "CONCENTRACION DE LOS ACIDOS" with an empty input field.
- A button labeled "ACEPTAR" below the first text box.
- A text box labeled "ACIDO ESPECIFICO" with an empty input field.
- A button labeled "CALCULAR" below the "ACIDO ESPECIFICO" text box.
- A button labeled "BORRAR" to the right of the "CALCULAR" button.
- A text box labeled "CANTIDAD DE BOTELLAS DEL ACIDO ESPECIFICO" with an empty input field.
- A button labeled "SALIR" below the second text box.
- A list box labeled "ListBox1" on the left side of the form.

3.- CODIFICACION

```
Public Conc(99) As Double
```

```
Public I As Integer
```

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button1.Click
```

```
    Conc(I) = TextBox1.Text
```

```
    ListBox1.Items.Add(CONC(I))
```

```
    I = I + 1
```

```
TextBox1.Text = ""
```

```
TextBox1.Focus()
```

```
End Sub
```

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button2.Click
```

```
Dim C, K As Integer
```

```
Dim Acesp As Double
```

```
K = I - 1
```

```
Acesp = TextBox2.Text
```

```
For I = 0 To K
```

```
    If Conc(I) = Acesp Then
```

```
        C = C + 1
```

```
    End If
```

```
Next
```

```
TextBox3.Text = C
```

```
End Sub
```

```
End Class
```



APLICACIÓN 19

1.- ENUNCIADO DEL PROBLEMA:

En una planta de producción de cosméticos, los productos se clasifican como aceptables y no aceptables de acuerdo al porcentaje de impurezas que contiene dicho cosmético, así como se muestra en la siguiente tabla:

% de impurezas	Categoría
$\leq 30 \%$	Aceptable
$>30\%$	Rechazo

Tabla - 14

Realice un programa en donde se pueda ingresar los porcentajes de impureza de n cosméticos y determine cuántos de estos se ubican en cada una de las categorías ya mencionadas.

DATO:

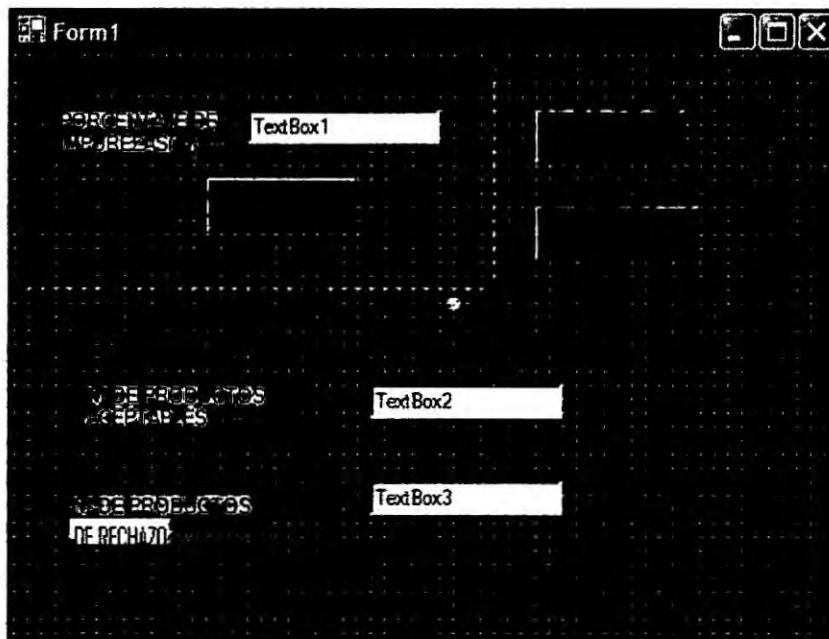
$LISTA_i$ = vector variable numérica entera, donde se almacenan los porcentajes de impurezas

INC = variable numérica entera, contador de porcentaje de impureza en la categoría de inaceptables

ACP = variable numérica entera, contador de porcentaje de impureza en la categoría de aceptables.

2.-

DISEÑO FORMULARIO-18



3.- CODIFICACION

```
Public I, INC, ACP, LISTA(100) As Integer
```

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button1.Click
```

```
    LISTA(I) = TextBox1.Text
```

```
    I = I + 1
```

```
    TextBox1.Clear()
```

```
    TextBox1.Focus()
```

```
End Sub
```

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button2.Click
```

```
    Dim K As Integer
```

```
    K = I - 1
```

```
    For I = 0 To K
```



```
If LISTA(I) >= 30 Then
```

```
    INC = INC + 1
```

```
Else
```

```
    ACP = ACP + 1
```

```
End If
```

```
Next
```

```
TextBox3.Text = INC
```

```
TextBox2.Text = ACP
```

```
End Sub
```

```
End Class
```



APLICACIÓN 20

1. ENUNCIADO DEL PROBLEMA:

Crear un programa donde al ir ingresando los potenciales de hidrogeno (pH) de cómo resultados cuantos son ácidos, bases y neutros, así como se muestra en la siguiente tabla:

pH	CLASIFICACION
≥ 0 y < 7	Acido
$= 7$	Neutro
> 7 AND ≤ 14	Básico

Tabla - 15

DATO

pH = variable tipo numérico real que define el potencial de hidrogeno

AC = variable tipo numérico entero contador de ácidos

NE = variable tipo numérico entero contador de neutro

BA = variable tipo numérico entero contador de básico

NROi = vector donde se almacena los pH, tipo numérico real $0 \leq \text{pH} \leq 14$.

2. DISEÑO FORMULARIO-19:

The image shows a window titled "POTENCIAL DE HIDROGENO (PH):". Inside the window, there is a text input field labeled "INGRESAR EL PH". To the right of this field are three buttons: "CALCULAR", "BORRAR", and "ACEPTAR". Below the input field, there are three labels: "ACIDOS", "NEUTRO", and "BASE(ALCALINO)", each followed by a horizontal line representing a counter. At the bottom right of the window is a "CERRAR" button. The window also features standard Windows-style window controls (minimize, maximize, close) in the top right corner.

3. CODIFICACION:

Public NRO(19), I As Integer

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

NRO(I) = TextBox1.Text

ListBox1.Items.Add(NRO(I))

TextBox1.Text = ""

TextBox1.Focus()

I = I + 1

End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click

Dim K, AC, NE, BA As Integer

K = I - 1

For I = 0 To K

If NRO(I) = 7 Then

NE = NE + 1

Elseif NRO(I) > 7 and NRO(I) ≤ 14 then

BA = BA + 1

End if

Next

TextBox2.Text = AC

TextBox3.Text = NE

TextBox4.Text = BA

End Sub



APLICACIÓN 21

1.- ENUNCIADO DEL PROBLEMA:

En el laboratorio queremos saber el volumen total de HCL 0.1M en ml. Si en cada tubo de ensayo se agrega una cantidad de sustancia de acuerdo al número de tubos:

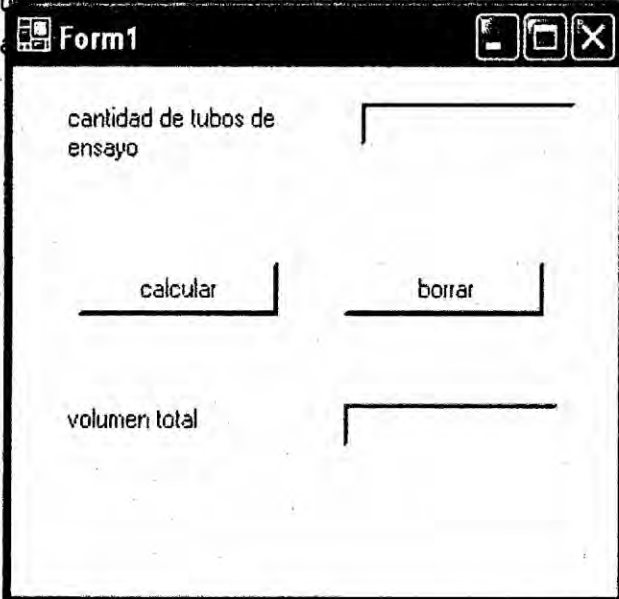
1er tubo → 1ml

2do tubo → 2ml y así sucesivamente

Realizar un programa que halle el volumen total de una cantidad "n" de tubos de ensayo.

$$S = 1 + 2 + 3 + 4 + \dots + n$$

2.- DISEÑO FORMULARIO-20



The image shows a screenshot of a Windows application window titled "Form1". The window contains a form with the following elements:

- A label "cantidad de tubos de ensayo" followed by an empty text input field.
- Two buttons: "calcular" and "borrar".
- A label "volumen total" followed by an empty text input field.

[Handwritten signature]

3.- CODIFICACION

```
Public Class Form1
```

```
    Inherits System.Windows.Forms.Form
```

```
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
```

```
System.EventArgs) Handles Button1.Click
```

```
        Dim s, c, n As Integer
```

```
        s = 0
```

```
        c = 1
```

```
        n = TextBox1.Text
```

```
        Do While c <= n
```

```
            s = s + c
```

```
            c = c + 1
```

```
        Loop
```

```
        TextBox2.Text = s
```

```
    End Sub
```

```
End Class
```



CAPITULO III

3.1 Arreglos Unidimensionales

También son llamados arrays unidimensionales y lo podríamos definir como un conjunto de variables del mismo tipo y tamaño que ocupan posiciones consecutivas en la memoria del ordenador. El tamaño en memoria que ocupa un array es siempre fijo y no puede variar. Para calcular el tamaño en memoria que puede ocuparnos un array solo tenemos que multiplicar el número de elementos de nuestro array por el tamaño en bytes del tipo de este.

La estructura más simple es el arreglo unidimensional, que consiste de una columna de localizaciones de memoria. El siguiente arreglo es un arreglo unidimensional llamado AGE. Los elementos dados del arreglo son similares a los nombres de referencia, dado que el primer elemento (con el dato 32) es la caja 1 (en lugar de 0) del arreglo. Los nombres de referencia se escriben como AGE(1), AGE(2), y así sucesivamente. El número entre paréntesis es sólo un número de referencia y puede ser una constante, una variable o una expresión.



Arreglo Unidimensional

	Array	Variable
	AGE	Reference Name
1	32	AGE(1)
2	54	AGE(2)
3	25	AGE(3)
4	36	AGE(4)
5	45	AGE(5)
6	20	AGE(6)
7	28	AGE(7)
8	50	AGE(8)
9	42	AGE(9)

El número entre paréntesis hace referencia o apunta al número de la caja en el arreglo, que es el número del elemento

Ejemplos Aplicativos:

APLICACIÓN 22

1.- ENUNCIADO DEL PROBLEMA:

Hacer un programa en visual Basic.net, y un diagrama de flujo, que reciba como entrada las N concentraciones de un ácido que se obtuvieron en varios experimentos, los almacene en vector unidimensional y luego nos permita calcular:

- La mayor concentración obtenida, y su posición.
- La menor concentración obtenida, y su posición
- El promedio de las concentraciones obtenidas.

Dato:

Conc(i)=Variable que almacena el valor de las concentraciones que se ingresa.

may = concentración mayor

men = concentración menor

Pos= Posición de la concentración mayor

Pos1= Posición de la concentración menor

Prom= Promedio de las concentraciones.

POSICION	CONCENTRACION (M)
1	0.0001
2	0.001
3	0.0015
4	0.002
5	0.01
6	0.018
7	0.02
8	0.1
9	0.16
10	0.3

Tabla - 16

2.- DISEÑO FORMULARIO-21

The image shows a software form window titled "Form1". The form is titled "CONCENTRACIONES OBTENIDAS". It contains the following elements:

- A label "CONCENTRACION" followed by a text input field.
- A "CALCULAR" button.
- A "BORRAR" button.
- An "ACEPTAR" button.
- A large text input field on the left side.
- Labels for "PROM DE CONCS", "MAYOR CONC", "POS", "MENOR CONC", and "POS1", each followed by a text input field.
- A "SALIR" button.

3.- CODIFICACION

Public I, CONC (99), SUMA As Double

Private Sub Button1_Click (ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

CONC (I) = TextBox1.Text

ListBox1.Items.Add (CONC (I))

I = I + 1

TextBox1.Text = ""

TextBox1.Focus ()

End Sub

Private Sub Button2_Click (ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click

Dim K, WMAY, PROM As Double

Dim POS, POS1 As Integer

Dim WMEN As Integer = 99999

K = I - 1

For I = 0 To K

If CONC (I) > WMAY Then

WMAY = CONC (I)

POS = I

End If

If CONC (I) < WMEN Then

WMEN = CONC (I)

POS1 = I

End If

SUMA = SUMA + CONC (I)

Next

PROM = SUMA / I

TextBox2.Text = PROM

TextBox3.Text = WMAY

TextBox4.Text = POS

TextBox5.Text = WMEN

TextBox6.Text = POS1

End Sub

End Class



APLICACIÓN 23

1.- ENUNCIADO DEL PROBLEMA:

Crear un programa en la cual nos calcule la cantidad de longitudes de onda que nos reflejan la coloración violeta, azul, verde, amarilla, naranja, rojo que hay al ingresar las longitudes de onda de la luz visible y además nos indique el mayor de ellos respecto de cada coloración, así como se muestra en la siguiente tabla:

LONGITUD DE ONDA	COLORACION
$400 \leq h \leq 446$	Violeta
$446 < h \leq 500$	Azul
$500 < h \leq 542$	Verde
$542 < h \leq 578$	Amarillo
$578 < h < 600$	Naranja
$600 < h \leq 700$	Rojo

Tabla - 17

2.- DISEÑO FORMULARIO-22

The screenshot shows a window titled 'Form1' with the main title 'LUZ VISIBLE'. At the top, there is a text input field labeled 'LONGITUD DE ONDA' and a 'CALCULAR' button to its right. Below the input field is an 'ACEPTAR' button. To the right of the 'ACEPTAR' button is a 'BORRAR' button. The main area of the form contains a list of colors: VIOLETA, AZUL, VERDE, NARANJA, AMARILLO, and ROJO. Each color name is followed by a text input field and the word 'MAYOR' followed by another text input field. At the bottom right of the form is a 'SALIR' button.

3.- CODIFICACION

Public Class Form1

Inherits System.Windows.Forms.Form

Public LO(11), I As Double

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

LO(I) = TextBox1.Text

ListBox1.Items.Add(LO(I))

I = I + 1

End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click

Dim K As Integer

Dim VIO, AZU, VER, NAR, AMA, ROJ, WMAY, WMAY2, WMAY3,
WMAY4, WMAY5, WMAY6 As Double

K = I - 1

For I = 0 To K

If LO(I) <= 446 And LO(I) >= 400 Then

VIO = VIO + 1

If LO(I) > WMAY Then

WMAY = LO (I)

End If

Elseif LO(I) <= 500 And LO(I) > 446 Then

AZU = AZU + 1

If LO(I) > WMAY2 Then

WMAY2 = LO(I)

End If

Elsif LO(I) <= 542 And LO(I) > 500 Then

VER = VER + 1

If LO(I) > WMAY3 Then

WMAY3 = LO (I)

End If

Elsif LO(I) <= 578 And LO(I) > 542 Then

AMA = AMA + 1

If LO(I) > WMAY4 Then

WMAY4 = LO(I)

End If

Elsif LO(I) <= 600 And LO(I) > 578 Then

NAR = NAR + 1

If LO(I) > WMAY5 Then

WMAY5 = LO(I)

End If

Elsif LO(I) <= 700 And LO(I) > 600 Then

ROJ = ROJ + 1

If LO(I) > WMAY6 Then

WMAY6 = LO (I)

End If



Else

VIO = VIO + 1

End If

Next

TextBox2.Text = VIO

TextBox3.Text = WMAY

TextBox4.Text = AZU

TextBox5.Text = WMAY2

TextBox6.Text = VER

TextBox7.Text = WMAY3

TextBox8.Text = AMA

TextBox9.Text = WMAY4

TextBox10.Text = NAR

TextBox11.Text = WMAY5

TextBox12.Text = ROJ

TextBox13.Text = WMAY6

End Sub

End Class



Tiempo (s)	Temperature (°C)
20	16
40	16.5
60	17
80	18.1
100	18.3
120	18.8
140	19
160	19.7
180	20.1
200	21.2

Tabla - 18

APLICACIÓN 24

1.- ENUNCIADO DEL PROBLEMA:

Escriba un programa en visual Basic tal que dado como entrada un arreglo unidimensional que contiene el pH de N soluciones calculadas en el laboratorio, determine el promedio y el numero de soluciones que tienen el pH inferior al promedio.

DATO:

pH(i): vector de tipo de variables numérica real, donde se almacena el pH $0 \leq$

$\text{pH}(i) \leq 14$

PROM: variable numérica real, calcula el promedio del pH

X: variable numérica real, contador de pH menor que l promedio

2.- DISEÑO FORMULARIO-23

The screenshot shows a Visual Basic form titled "Form1" with a list box containing the following pH values: 2.5, 6, 1, 3.2, 4, 6.9, 4.7, 1.1, 2.8, 6.3, and 3.5. To the right of the list box, there are two labels: "PROMEDIO" and "CANT. DEPH< PROMEDIO". The "PROMEDIO" label is positioned above a text box containing the value "4.2". The "CANT. DEPH< PROMEDIO" label is positioned above a text box containing the value "8". There are four buttons: "CALCULAR" (top right), "BORRAR" (middle right), "ACEPTAR" (middle left), and "SALIR" (bottom right). The "PH" label is positioned above a text box that is currently empty.

3.- CODIFICACION

Public Class Form1

Inherits System.Windows.Forms.Form

Public PH(49), I, S As double

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As

System.EventArgs) Handles Button1.Click

 PH(I) = TextBox1.Text

 ListBox1.Items.Add(PH(I))

 S = S + PH(I)

 TextBox1.Text = " "

 TextBox1.Focus ()

 I = I + 1

End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As

System.EventArgs) Handles Button2.Click

 Dim PROM As Double

 Dim X, K As Integer

 PROM = S / I

 K = I - 1

 For I = 0 To K

 If PH(I) < PROM Then

 X = X + 1

 End If

 Next

```
    TextBox2.Text = PROM
```

```
    TextBox3.Text = X
```

```
End Sub
```

```
End Class
```



APLICACIÓN 25

1.- ENUNCIADO DEL PROBLEMA:

Desarrollar un programa en visual basic que le permita calcular el calor integral de reacción, durante un experimento de calorimetría, donde se registran temperaturas cada cierto intervalo de tiempo.

Datos

Estas son las variables a utilizar:

CC = Capacidad calorífica

M = masa de la solución

CP = calor a presión constante

NRO(I) = vector que almacena las temperaturas

QINT = calor integral

$QINT = (CC + M) \times CP \times (T_i - T_f)$

2.- DISEÑO FORMULARIO-24

The image shows a screenshot of a Visual Basic form titled "CALOR INTEGRAL". The form is divided into several sections:

- Top Section:** Contains three input fields labeled "CAPACIDAD CALORIFICA", "MASA DE SOLUCION", and "CP". To the right of these fields is a button labeled "ACEPTAR".
- Middle Section:** Contains one input field labeled "TEMPERATURAS REGISTRADAS" and a button labeled "ACEPTAR".
- Bottom Section:** Contains four buttons: "CALOR INTEGRAL" (with an adjacent input field), "MOSTRAR", "TEMPERATURA MAYOR" (with an adjacent input field), "BORRAR", and "SALIR".

CAPITULO IV

4.1 Arreglos Bidimensionales (matrices)

Un array bidimensional (también llamado tabla o matriz) es un array con dos índices. Al igual que los vectores deben ser ordinales. Se declaran de igual manera que los arrays de una dimensión.

Un array bidimensional recoge valores de una tabla de doble entrada. Cada uno de los elementos se identifica y se asigna mediante una variable (*\$nombre*) seguida de dos () que contienen los *índices* del array. Los índices puede ser escalares -equivaldrían al número de fila y columna que la celda ocuparía en la tabla, o puede ser asociativo que equivaldría en alguna medida a usar como índices *los nombres de la fila y de la columna*

Un array bidimensional (tabla o matriz) es un array con dos índices, al igual que los vectores que deben ser ordinales o tipo subrango.

		Columnas				
		1	2	3	4	5
Filas	1	A[1,1]	A[1,2]	A[1,3]	A[1,4]	A[1,5]
	2					
	3					
	4	A[4,1]				A[4,5]

Para localizar o almacenar un valor en el array se deben especificar dos posiciones (dos subíndices), uno para la fila y otro para la columna.

Formato:

- *identificador = array [índice1, índice 2] of tipo de elemento*
- *identificador = array [índice 1] of array [índice 2] of tipo de elemento*



EJEMPLOS APLICATIVOS

APLICACION 26

1.- ENUNCIADO DEL PROBLEMA:

Crear una matriz que me permita almacenar la cantidad de átomos de carbono e hidrógeno presentes en un hidrocarburo y que luego muestre la cantidad de alcanos, alquenos y alquinos ingresados. (Alcanos, alquenos y alquinos son tipos de hidrocarburos)

Dato:

HIDRO (1...9; 1...2)

HIDRO es un arreglo bidimensional de tipo entero que almacena la cantidad de átomos de carbono e hidrógeno de nueve hidrocarburos

(Alcano= C_NH_{2N+2} ; Alqueno= C_NH_{2N} ; Alquino= C_NH_{2N-2})

HIDROCARBUROS		(1) N° de átomos de Hidrógeno	(2) N° de átomos de carbono
	1		
	2		
	3		
	...		
	9		

2.- CODIFICACIÓN:

Module Module1

Sub Main()

Dim HIDRO (9, 2) As Integer

Dim F, C, X1, X2, X3 As Integer

Dim TIPO As String

For F = 1 To 9

For C = 1 To 2

Console.WriteLine("INGREAR FILA==>" & F & "COLUMNA==>" & C)

HIDRO(F, C) = Console.ReadLine

Next

Next

For F = 1 To 9

If HIDRO(F, 2) = 2 * (HIDRO(F, 1)) + 2 Then

TIPO = "ALCANO"

X1 = X1 + 1

End If

If HIDRO(F, 2) = 2 * (HIDRO(F, 1)) Then

TIPO = "ALQUENO"

X2 = X2 + 1

End If

If HIDRO(F, 2) = 2 * (HIDRO(F, 1)) - 2 Then

TIPO = "ALQUINO"

X3 = X3 + 1

End If

Next

Console.WriteLine(" LA CANTIDAD DE ALCANOS ES: " & X1)

Console.WriteLine(" LA CANTIDAD DE ALQUENOS ES: " & X2)

Console.WriteLine(" LA CANTIDAD DE ALQUINOS ES: " & X3)

Console.ReadLine()

End Sub

End Module



APLICACION 27

1.- ENUNCIADO DEL PROBLEMA:

Diseñe un programa que indique la acidez o basicidad de una solución a partir de los colores que toman ciertos indicadores en dicha solución, así como se muestra en la siguiente tabla:

	Tornasol	Fenolftaleina	Rojo de Metilo
Solucion1	Color 1,1	Color 1,2	Color 1,3
Solucion2	Color 2,1	Color 2,2	Color 2,3
Solucion3	Color 3,1	Color 3,2	Color 3,3

Tabla - 19

Datos:

Color i j: Es una variable tipo cadena que indica el color que toma la solución con cierto indicador

2.- CODIFICACION

Module Module1

Sub Main()

Dim color(6, 2), medio, ind(2) As String

Dim f, c As Integer

For f = 0 To 6

For c = 0 To 2

color(f, c) = Console.ReadLine()

```

If c = 0 Then
    ind(c) = "tornasol"
Elseif c = 1 Then
    ind(c) = "fenolftaleina"
Elseif c = 2 Then
    ind(c) = "rojo de metilo"
End If

Console.WriteLine("Solucion N°==>" & f + 1 & "
                    Indicador==> " & ind(c))

If c = 0 Then
    If color(f, 0) = "rojo" Then
        Console.WriteLine("El medio es acido")
    Elseif color(f, 0) = "azul" Then
        Console.WriteLine("El medio es basico")
    End If
End If

If c = 1 Then
    If color(f, 1) = "incoloro" Then
        Console.WriteLine("El medio es acido")
    Elseif color(f, 1) = "rojo" Then
        Console.WriteLine("El medio es basico")
    End If
End If

If c = 2 Then

```

```
If color(f, 2) = "rojo" Then
    Console.WriteLine("El medio es acido")
Elseif color(f, 2) = "amarillo" Then
    Console.WriteLine("El medio es basico")
End If
End If
Next
Console.Read()
Next
End Sub
End Module
```



APLICACIÓN 28

1.- ENUNCIADO DEL PROBLEMA:

En un laboratorio de industria química, el año anterior se almacenaron las concentraciones de 8 ácidos distintos en un mes.

Datos:

CLAVE (i): una variable de tipo entero, que representa el código de ácidos

CONC i, j : una variable de tipo real, que representa la concentración del ácido i en el día j .

Nos piden:

- a) Clave del ácido con mas alta concentración en el mes y su respectiva concentración
- b) El promedio de concentración de cada ácido.

Clave (i)

Código de ácidos:

5	7	
---	---	-------	--

$\begin{matrix} 7 \\ \underbrace{\hspace{1.5cm}} \\ n-1 \end{matrix}$

Solo falta aumentar más códigos de ácidos y concentración del ácido uno al día, y todo está bien.

Con $i, j \rightarrow$ concentración

	0	1		
0	20	25		
1	30	35		
.				
.				
.				
N-1				

$\begin{matrix} 7 \\ \underbrace{\hspace{1.5cm}} \\ n-1 \end{matrix}$



2.- DISEÑO FORMULARIO-25

```
C:\Documents and Settings\pok 000012\Mis documentos\Visual Studio Projects\ConsoleApp...
INGRESE ACIDO0
5
INGRESE ACIDO1
?
INGRESE LA CONCENTRACION DEL MES
INGRESE LA CONCENTRACION FILA ->0 COLUMNA ->0
20
INGRESE LA CONCENTRACION FILA ->0 COLUMNA >1
25
INGRESE LA CONCENTRACION FILA =>1 COLUMNA =>0
30
INGRESE LA CONCENTRACION FILA ->1 COLUMNA ->1
35
DETERMINE LA MAYOR CONCENTRACION
LA CONCENTRACION ES >35
LA CLAVE DEL ACIDO ES ->7
EL PROMEDIO DE CONCENTRACION DE CADA ACIDO ES ->22,5
EL PROMEDIO DE CONCENTRACION DE CADA ACIDO ES =>32,5
-
```

3.- CODIFICACION MODO CONSOLA

Module Module1

Sub Main()

Dim CONC (7, 30), CLAVE(7), WMAY, SUMA, I, F, C, WCLAVE As

Integer

Dim PROM As Double

For I = 0 To 1

Console.WriteLine("INGRESE ACIDO" & I)

CLAVE (I) = Console.ReadLine

Next

Console.WriteLine("INGRESE LA CONCENTRACION DEL MES")

For F = 0 To 1

For C = 0 To 1

```

    Console.WriteLine("INGRESE LA CONCENTRACION FILA =>" & F &
" COLUMNA =>" & C)
    CONC (F, C) = Console.ReadLine
    Next
Next
Console.WriteLine("DETERMINE LA MAYOR CONCENTRACION")
For F = 0 To 1
    For C = 0 To 1
        If CONC(F, C) > WMAY Then
            WMAY = CONC(F, C)
            WCLAVE = F
        End If
    Next
Next
Console.WriteLine("LA CONCENTRACION ES=>" & WMAY)
Console.WriteLine("LA CLAVE DEL ACIDO ES =>" & CLAVE(WCLAVE))
For F = 0 To 1
    For C = 0 To 1
        SUMA = SUMA + CONC(F, C)
    Next
    PROM = SUMA / C
    Console.WriteLine("EL PROMEDIO DE CONCENTRACION DE CADA
ACIDO ES =>" & PROM)
    PROM = 0
    SUMA = 0

```

Next

Console.ReadLine()

End Sub

End Module



APLICACIÓN 29

1.- ENUNCIADO DEL PROBLEMA:

En un laboratorio de química; en un día se realiza 5 experimentos, donde se encontró distintos volúmenes de un gas desconocido

Datos:

Tem $i \rightarrow$ Es una variable de tipo entero, que representa la temperatura del día.

Vol $i, j \rightarrow$ Es una variable de tipo real, que representa el volumen del día i en el experimento j .

Si en dicho laboratorio se desea saber:

- a) El número del experimento donde se encuentra el mayor volumen.

La temperatura en el que se encontró.

El mayor volumen.

- b) El día donde se encuentra el volumen menor.

El volumen menor.

- c) El promedio diario de volumen más alto.

SOLUCION

Tem i

0	1	N-1
19	22		

Vol i, j

	0	1	4
0	28	23		
1	15	21		
			
			.	
			.	
			.	
N-1				

2.- DISEÑO FORMULARIO-26 EN MODO CONSOLA

```
C:\Documents and Settings\pok 000012\Mis documentos\Visual Studio Projects\ConsoleApp - [ ] X
INGRESE LA CANTIDAD DE DIAS
2
INGRESE TEMPERATURA DEL EXPERIMENTO0
19
INGRESE TEMPERATURA DEL EXPERIMENTO1
22
INGRESE LOS VOLUMENES OBTENIDOS
INGRESE VOLUMEN FILA => 0 COLUMNA => 0
28
INGRESE VOLUMEN FILA => 0 COLUMNA -> 1
23
INGRESE VOLUMEN FILA => 1 COLUMNA -> 0
15
INGRESE VOLUMEN FILA => 1 COLUMNA => 1
21
EL VOLUMEN MAYOR ES => 28
EL NUMERO DE EXPERIMENTO DONDE SE OBTUVO EL VOLUMEN MAYOR ES => 0
SU TEMPERATURA ES => 19
EL VOLUMEN MENOR ES => 15
EL DIA DONDE SE OBTUVO EL VOLUMEN MENOR ES ->0
EL MAYOR PROMEDIO DIARIO ES => 25,5
```

3.- CODIFICACION

Module Module1

Sub Main ()

Dim N, I, F, C, WDIA, WNUM, SUMA, VOL (30, 4), TEM (30), WMAY,

WMEN, WTEM as Integer

Dim PROM, WM as Double

WMEN = 999999

Console.WriteLine ("INGRESE LA CANTIDAD DE DIAS")

N = Console.ReadLine

For I = 0 to N - 1

Console.WriteLine ("INGRESE TEMPERATURA DEL EXPERIMENTO"

& I)

TEM (I) = Console.ReadLine

Next

```

Console.WriteLine ("INGRESE LOS VOLUMENES OBTENIDOS")
For F = 0 to N - 1
    For C = 0 to N - 1
        Console.WriteLine (" INGRESE VOLUMEN FILA => " & F & "
COLUMNNA => " & C)
        VOL (F, C) = Console.ReadLine
    Next
Next
For F = 0 to N - 1
    For C = 0 to N - 1
        If VOL (F, C) > WMAY Then
            WMAY = VOL (F, C)
            WTEM = F
            WNUM = F
        End If
        If VOL (F, C) < WMEN Then
            WMEN = VOL (F, C)
            WDIA = C
        End If
    Next
Next
Console.WriteLine ("EL VOLUMEN MAYOR ES => " & WMAY)
Console.WriteLine (" EL NUMERO DE EXPERIMENTO DONDE SE
OBTUVO EL VOLUMEN MAYOR ES => " & WNUM)
Console.WriteLine ("SU TEMPERATURA ES => " & TEM(WTEM))

```

```
Console.WriteLine ("EL VOLUMEN MENOR ES => " & WMEN)

Console.WriteLine ("EL DIA DONDE SE OBTUVO EL VOLUMEN MENOR
ES =>" & WDIA)

For F = 0 to N - 1
    For C = 0 to N - 1
        SUMA = SUMA + VOL (F, C)
    Next
    PROM = SUMA / C
    If PROM > WM Then
        WM = PROM
    End If
    SUMA = 0
    PROM = 0
Next

Console.WriteLine ("EL MAYOR PROMEDIO DIARIO ES => " & WM)

Console.ReadLine ()

End Sub

End Module
```



CAPITULO V

5.1 Procedimientos

Los procedimientos son muy interesantes y útiles en la programación. Nos sirven para realizar una tarea concreta que probablemente se vaya a ejecutar varias veces a lo largo de la vida de la página. Esta tarea se especifica en un bloque de código de manera independiente y cuando se desean realizar las acciones del procedimiento se llama al procedimiento o función. Una vez realizadas las acciones pertinentes se devuelve el flujo del programa al lugar desde donde se invocó ese procedimiento.

Lo primero que debemos hacer al crear un procedimiento es pensar las cosas que se desean hacer dentro de la función, la información que necesitaremos (y que tendremos que recibir como parámetros) y la información que devolverá. Con estas ideas claras se pueden construir los procedimientos y funciones sin mucha dificultad, siguiendo estas estructuras.

Para un procedimiento

Sub nombre (parametro1, parametro2...)

... Código del procedimiento

end Sub



Ejemplos Aplicativos

APLICACION 30

1.- ENUNCIADO DEL PROBLEMA:

Determinación de la Densidad

Datos

M = Masa

V = Volumen

D = Densidad

$D = M / V$

DenSidad = Nombre del procedimiento

2.- DISEÑO FORMULARIO-27

Form1

DeTerMiNaCioN De La DenSiDaD™

MASA gr.

VOLUMEN ml.

CALCULAR BORRAR

DENSIDAD g/ml.

3.- CODIFICACION

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

Dim m, v, d As Double

m = TextBox1.Text

v = TextBox2.Text

DenSiDad_(m, v, d)

End Sub

Sub DenSiDad_(ByVal m As Double, ByVal v As Double, ByRef d As Double)

d = m / v

TextBox3.Text = d

End Sub

End Class



APLICACION 31

1.- ENUNCIADO DEL PROBLEMA:

Tabla de Conversión de La Temperatura

Datos

$^{\circ}\text{F}$ = Grados Fahrenheit

$^{\circ}\text{C}$ = Grados Centígrados

$^{\circ}\text{K}$ = Grados Kelvin

$$^{\circ}\text{F a } ^{\circ}\text{C} = (^{\circ}\text{T} - 32) / 1.8$$

$$^{\circ}\text{C a } ^{\circ}\text{F} = 1.8 * ^{\circ}\text{T} + 32$$

$$^{\circ}\text{K a } ^{\circ}\text{C} = ^{\circ}\text{T} - 273.15$$

$$^{\circ}\text{C a } ^{\circ}\text{K} = ^{\circ}\text{T} + 273.15$$

$$^{\circ}\text{F a } ^{\circ}\text{K} = (100 * (^{\circ}\text{T} - 32) / 180) + 273.15$$

$$^{\circ}\text{K a } ^{\circ}\text{F} = (180 * (^{\circ}\text{T} - 273.15) / 100) + 32$$

2.- DISEÑO FORMULARIO-28

Form1

« Tabla D' Conversion D' Temperaturas »

Temperatura

Hazle Click Al Que Quieras

$^{\circ}\text{C a } ^{\circ}\text{K}$	<input type="text"/>	$^{\circ}\text{K a } ^{\circ}\text{C}$	<input type="text"/>
$^{\circ}\text{C a } ^{\circ}\text{F}$	<input type="text"/>	$^{\circ}\text{F a } ^{\circ}\text{C}$	<input type="text"/>
$^{\circ}\text{K a } ^{\circ}\text{F}$	<input type="text"/>	$^{\circ}\text{F a } ^{\circ}\text{K}$	<input type="text"/>

3.- CODIFICACION

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

Dim t, CaK As Double

t = TextBox1.Text

C_K(t, CaK)

End Sub

Sub C_K(ByVal t As Double, ByRef CaK As Double)

CaK = t + 273

TextBox2.Text = CaK

End Sub

Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button3.Click

Dim t, KaC As Double

t = TextBox1.Text

K_C(t, KaC)

End Sub

Sub K_C(ByVal t As Double, ByRef KaC As Double)

KaC = t - 273

TextBox4.Text = KaC

End Sub



Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click

Dim t, CaF As Double

t = TextBox1.Text

C_F(t, CaF)

End Sub

Sub C_F(ByVal t As Double, ByRef CaF As Double)

CaF = 1.8 * t + 32

TextBox3.Text = CaF

End Sub

Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button4.Click

Dim t, FaC As Double

t = TextBox1.Text

F_C(t, FaC)

End Sub

Sub F_C(ByVal t As Double, ByRef FaC As Double)

FaC = (t - 32) / 1.8

TextBox5.Text = FaC

End Sub



Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button6.Click

Dim t, KaF As Double

t = TextBox1.Text

K_F(t, KaF)

End Sub

Sub K_F(ByVal t As Double, ByRef KaF As Double)

$KaF = ((180 * (t - 273)) / 100) + 32$

TextBox7.Text = KaF

End Sub

Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button5.Click

Dim t, FaK As Double

t = TextBox1.Text

F_K(t, FaK)

End Sub

Sub F_K(ByVal t As Double, ByRef FaK As Double)

$FaK = (100 * (t - 32) / 180) + 273$

TextBox6.Text = FaK

End Sub

End Class



APLICACION 32

1.- ENUNCIADO DEL PROBLEMA:

Determinación de la Molaridad, Normalidad, Molalidad:

DATOS

M = Masa del soluto

Mm = Masa molar

V = Volumen solución

Q = Equivalente / mol

Ms = Masa soluto

$$\text{Molaridad} = (m / Mm) * (1 / V)$$

$$\text{Normalidad} = (m \times Q) / (Mm \times V)$$

$$\text{Molalidad} = (m / Mm) * (1 / Ms)$$

2.- DISEÑO FORMULARIO-29

Masa Del SolutO	Masa Molar	Volumen de la Solución
<input type="text"/> Gr.	<input type="text"/> gr/mol	<input type="text"/> L.
θ	Masa Del Solvente	
<input type="text"/> eq/mol	<input type="text"/> Kg	
BORRAR		
MoLaRiDaD	<input type="text"/> mol/L	
NorMaLiDaD	<input type="text"/> Eq/L	
MoLaLiDaD	<input type="text"/> mol/kg	

3.- CODIFICACION

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
```

```
Dim m, Mm, V, Molaridad As Double
```

```
m = TextBox1.Text
```

```
Mm = TextBox2.Text
```

```
V = TextBox3.Text
```

```
MolaRiDaD_(m, Mm, V, Molaridad)
```

```
End Sub
```

```
Sub MolaRiDaD_(ByVal m As Double, ByVal Mm As Double, ByVal V As Double, ByRef Molaridad As Double)
```

```
Molaridad = (m / Mm) * 1 / V
```

```
TextBox6.Text = Molaridad
```

```
End Sub
```

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
```

```
Dim m, Mm, V, NorMalidad, Q As Double
```

```
m = TextBox1.Text
```

```
Mm = TextBox2.Text
```

```
V = TextBox3.Text
```

```
Q = TextBox4.Text
```

```
NorMaLiDaD_(m, Mm, V, Q, NorMalidad)
```

```
End Sub
```



```
Sub NorMaLiDaD_(ByVal m As Double, ByVal Q As Double, ByVal Mm As Double, ByVal V As Double, ByRef NorMalidad As Double)
```

```
    NorMalidad = (TextBox1.Text * TextBox4.Text) / (TextBox2.Text * TextBox3.Text)
```

```
    TextBox7.Text = NorMalidad
```

```
End Sub
```

```
Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button3.Click
```

```
    Dim m, Mm, MSTE, MOlaliDAD As Double
```

```
    m = TextBox1.Text
```

```
    Mm = TextBox2.Text
```

```
    MSTE = TextBox5.Text
```

```
    MolaLiDaD_(m, Mm, MSTE, MOlaliDAD)
```

```
End Sub
```

```
Sub MolaLiDaD_(ByVal m As Double, ByVal Mm As Double, ByVal MSTE As Double, ByRef MoLALIDAD As Double)
```

```
    MoLALIDAD = (m / Mm) * (1 / MSTE)
```

```
    TextBox8.Text = MoLALIDAD
```

```
End Sub
```

```
End Sub
```

```
End Class
```



APLICACION 33

1.- ENUNCIADO DEL PROBLEMA:

En una industria química se desea crear un programa que permita hallar el numero de moles y la molaridad de una determinada sustancia, para si facilitar el trabajo d e los ingenieros químicos.

Datos:

P = Peso molecular

W = Masa de la sustancia

V = Volumen de la solución

N = Numero de Moles

M = Molaridad

$$N = W / P \quad M = N / (P * V)$$

2.- DISEÑO FORMULARIO-30

The image shows a screenshot of a software application window titled "Form1". The window contains several input fields and buttons for calculating moles and molarity. The fields are labeled as follows:

- PESO MOLECULAR**: Input field with the value "50" and the unit "g/mol".
- masa**: Input field with the value "25" and the unit "gramos".
- VOLUMEN**: Input field with the value "0,2" and the unit "Litros".
- NUMERO DE MOLES**: Input field with the value "0,5" and the unit "mol".
- MOLARIDAD**: Input field with the value "2,5" and the unit "mol/L".

Buttons are located to the right of the input fields:

- A "calcular" button is positioned to the right of the "PESO MOLECULAR" field.
- A "borrar" button is positioned to the right of the "VOLUMEN" field.
- A "salir" button is located at the bottom right of the form.

3.- CODIFICACION

Public Class Form1

Inherits System.Windows.Forms.Form

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As

System.EventArgs) Handles Button1.Click

Dim P, W, V, N, M As Double

P = TextBox1.Text

W = TextBox2.Text

V = TextBox3.Text

TextBox1.Focus ()

TextBox1.text=""

CALCULAR_MOLES_MOLARIDAD(P, W, V, N, M)

TextBox4.Text = N

TextBox5.Text = M

End Sub

Sub CALCULAR_MOLES_MOLARIDAD(ByVal P As Double, ByVal W As

Double, ByVal V As Double, ByRef N As Double, ByRef M As Double)

$N = (W / P)$

$M = W / (P * V)$

End Sub

End Class



APLICACIÓN 34

1.- ENUNCIADO DEL PROBLEMA:

Realizar un programa el cual indique el equivalente gramo y el número equivalente gramo de una reacción, que se usa con frecuencia en el laboratorio de química. Teniendo en cuenta la masa, el peso atómico y electrones transferidos de una reacción.

DATO:

M = Variable tipo numérica real, lo cual representa la masa

Pa= Variable tipo numérica real, lo cual representa el peso atómico

EI = Variable tipo numérica entero, lo cual representa electrones transferidos

Ne= Variable tipo numérica real, número equivalente

Eg= Variable tipo numérica real, equivalente en gramo

Calcular= nombre del procedimiento.

$Eg = PA/EI$

$Ne = H/Eg$

2.- DISEÑO FORMULARIO-31

The image shows a screenshot of a Windows application window titled "Form1". The window contains a form with the following elements:

- Input field for "MASA"
- Input field for "PESO ATOMICO"
- Input field for "ELECTRONES DISOCIADOS"
- Button labeled "CALCULAR"
- Button labeled "BORRAR"
- Input field for "EQUIVALENTE GRAMO"
- Input field for "NUMERO EQUIVALENTE"
- Button labeled "SALIR"

3.- CODIFICACION

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles Button1.Click
```

```
Dim M, PA, EG, NE As Double
```

```
Dim EL As Integer
```

```
M = TextBox1.Text
```

```
PA = TextBox2.Text
```

```
EL = TextBox3.Text
```

```
CALCULAR(M, PA, EL, NE, EG)
```

```
TextBox4.Text = EG
```

```
TextBox5.Text = NE
```

```
End Sub
```

```
Sub CALCULAR(ByVal M As Double, ByVal PA As Double, ByVal EL As  
Integer, ByRef NE As Double, ByRef EG As Double)
```

```
EG = PA / EL
```

```
NE = M / EG
```

```
End Sub
```

```
End Class
```



CAPITULO VI

6.1 FUNCIONES

Una función en visual basic net es un modulo de un programa separado del cuerpo principal, que realiza una tarea específica y que puede regresar un valor a la parte principal del programa u otra función o procedimiento que la invoque.

La forma general de una función es:

Function Nom_fun(parametros)

instrucciones

nomfun = cargarlo porque es quien regresa el dato

End Function

La lista de parámetros formales es una lista de variables separadas por comas (,) que almacenaran los valores que reciba la función estas variables actúan como locales dentro del cuerpo de la función.

Aunque no se ocupen parámetros los paréntesis son requeridos.

Dentro del cuerpo de la función deber haber una instrucción que cargue el **NOMFUNCION** para regresar el valor, de esta manera se regresan los datos.

Sin embargo es de considerar que **NOMFUNCION** puede regresar un dato, una variable o una expresión algebraica (no ecuación o formula)



Ejemplos Aplicativos

APLICACIÓN 35

1.- ENUNCIADO DEL PROBLEMA:

Diseñar un programa utilizando Funciones, que nos de el tipo de enlace químico según el tipo de cristal indicado como dato.

Dato:

NRO	TIPO DE CRISTAL	TIPO DE ENLACE	PROPIEDADES
1	IONICO	Iónico	Alta dureza
2	COVALENTE	Covalente	Elevadísima dureza
3	MOLECULAR	Intermoleculares	Baja dureza
4	METALICO	Metálico	Dureza variable

Tabla - 20

2.- DISEÑO FORMULARIO-32

Form1

NRO. DE CRISTAL

CALCULAR BORRAR

TIPO DE CRISTAL

TIPO DE ENLACE QUIMICO

PROPIEDADES

SALIR

3.- CODIFICACION

Public Class Form1

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

Dim cristales As Integer

Dim Tip, Enlace, Prop As String

cristales = TextBox1.Text

Tip = Tipocristal(cristales)

TextBox2.Text = Tip

Enlace = Enlacequimico_cristal(cristales)

TextBox3.Text = Enlace

Prop = Propiedades_cristal(cristales)

TextBox4.Text = Prop

End Sub

Function Tipocristal(ByVal cristales As Integer) As String

Dim Tip As String

Select Case cristales

Case 1 : Tip = "IONICO"

Case 2 : Tip = "COVALENTE"

Case 3 : Tip = "MOLECULAR"

Case 4 : Tip = "METALICO"

Case Else

Tip = "ERROR"

End Select

Tipocristal = Tip



End Function

Function Enlacequimico_cristal(ByVal cristales As Integer) As String

Dim Enlace As String

Select Case cristales

Case 1 : Enlace = "ENLACE IONICO"

Case 2 : Enlace = "ENLACE COVALENTE"

Case 3 : Enlace = "ENLACES INTERMOLECULARES"

Case 4 : Enlace = "ENLACE METALICO"

Case Else

Enlace = "ERROR"

End Select

Enlacequimico_cristal = Enlace

End Function

Function Propiedades_cristal(ByVal cristales As Integer) As String

Dim Prop As String

Select Case cristales

Case 1 : Prop = "ALTA DUREZA"

Case 2 : Prop = "ELEVADISIMA DUREZA"

Case 3 : Prop = "BAJA DUREZA"

Case 4 : Prop = "DUREZA VARIABLE"

Case Else

Prop = "ERROR"

End Select

Propiedades_cristal = Prop

End Function



APLICACIÓN 36

1.- ENUNCIADO DEL PROBLEMA:

Hacer un programa en visual basic, que devuelva el nombre del hidrocarburo, que en este caso es el alcano y que para ello considere la cantidad de carbonos presente en dicho alcano, además devuelva el estado físico del mismo, así como se muestra en la siguiente tabla:

Dato:

Átomos de carbono:	Nombres IUPA	Estado Físico
C ₁ -C ₄	Metano Etano Propano Butano	Gases
C ₅ -C ₁₇	Pentano hexano tridecano Hexano decano tetradecano Heptano undecano pentadecano Octano hexadecano Heptadecano dodecano	Líquidos
C ₁₈ -C ₂₀	Octadecano Nonadecano Eicosano	Sólidos

Tabla - 20

2.- DISEÑO FORMULARIO-33



3.- CODIFICACION

Public Class Form1

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As

System.EventArgs) Handles Button1.Click

Dim atomo As Integer

Dim NOMB, Estado As String

atomo = TextBox1.Text

NOMB = Nombrelupac_atom(atomo)

TextBox2.Text = NOMB

Estado = Estadofisico_atom(atomo)

TextBox3.Text = Estado

End Sub

Function Nombrelupac_atom(ByVal atomo As Integer) As String

Dim Nombrelupac As String

Select Case atomo

Case 1 : Nombrelupac = "METANO"

Case 2 : Nombrelupac = "ETANO"

Case 3 : Nombrelupac = "PROPANO"

Case 4 : Nombrelupac = "BUTANO"

Case 5 : Nombrelupac = "PENTANO"

Case 6 : Nombrelupac = "HEXANO"

Case 7 : Nombrelupac = "HEPTANO"

Case 8 : Nombrelupac = "OCTANO"

Case 9 : Nombrelupac = "NONANO"

Case 10 : Nombrelupac = "DECANO"

Case 11 : Nombrelupac = "UNDECANO"

Case 12 : Nombrelupac = "DODECANO"

Case 13 : Nombrelupac = "TRIDECANO"

Case 14 : Nombrelupac = "TETRADECANO"

Case 15 : Nombrelupac = "PENTADECANO"

Case 16 : Nombrelupac = "HEXADECANO"

Case 17 : Nombrelupac = "HEPTADECANO"

Case 18 : Nombrelupac = "OCTADECANO"

Case 19 : Nombrelupac = "NONANODECANO"

Case 20 : Nombrelupac = "EICOSANO"

Case Else

Nombrelupac = "ERROR"

End Select

Nombrelupac_atom = Nombrelupac

End Function

Function Estadofisico_atom(ByVal atomo As Integer) As String

Dim Estado As String

Select Case atomo

Case 1, 2, 3, 4 : Estado = "GAS"

Case 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17 : Estado = "LIQUIDO"

Case 18, 19, 20 : Estado = "SOLIDO"

Case Else

Estado = "ERROR"

End Select

Estadofisico_atom = Estado

End Function

End Class



APLICACIÓN 37

1.- ENUNCIADO DEL PROBLEMA:

Desarrollar un programa basado en las funciones de la tabla periódica, que al ingresar un número de electrones devuelva el nombre, símbolo y peso atómico de los gases nobles.

DATOS

Gases nobles:

Helio (He) = 2 electrones

NOMBRE DEL ELEMENTO	SIMBOLO	NRO. ELECTRONES	PESO ATOMICO
Helio	(He)	02 electrones	2
Neon	(Ne)	10 electrones	20.18
Argon	(Ar)	18 electrones	40
Krypton	(Kr)	36 electrones	83.8
Xenon	(Xe)	54 electrones	131.29
Radon	(Rn)	86 electrones	222

Tabla - 21

ELEC= variable tipo numérica entera, nos indica el numero de electrones

NOMBRE= variable tipo strine, nos indica el numero de electrón

SIMBOLO= variable tipo strine, nos indica el numero de símbolo

PESO= variable tipo strine, nos indica el numero de electrón

2.- DISEÑO FORMULARIO-34

Form1

LOS GASES NOBLES

NUMERO DE ELECTRONES

ACEPTAR

BORRAR

ELEMENTO

SIMBOLO

PESO ATOMICO

SALIR

3.- CODIFICACION

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As

System.EventArgs) Handles Button1.Click

Dim ELEC As Integer

Dim NOM, SIM, PA As String

ELEC = TextBox1.Text

NOM = NOMBRE_ELE(ELEC)

TextBox2.Text = NOM

SIM = SIMBOLO_ELE(ELEC)

TextBox3.Text = SIM

PA = PESO_ATOMICO(ELEC)

TextBox4.Text = PA

End Sub

Function NOMBRE_ELE(ByVal ELEC As Integer) As String

Dim NOMBRE As String

Select Case ELEC

Case 2 : NOMBRE = "HELIO"

Case 10 : NOMBRE = "NEON"

Case 18 : NOMBRE = "ARGON"

Case 36 : NOMBRE = "KRIPTON"

Case 54 : NOMBRE = "XENON"

Case 86 : NOMBRE = "RADON"

Case Else

NOMBRE = "NO ES GAS NOBLE"

End Select

NOMBRE_ELE = NOMBRE

End Function

Function SIMBOLO_ELE(ByVal ELEC As Integer) As String

Dim SIMBOLO As String

Select Case ELEC

Case 2 : SIMBOLO = "He"

Case 10 : SIMBOLO = "Ne"

Case 18 : SIMBOLO = "Ar"

Case 36 : SIMBOLO = "Kr"

Case 54 : SIMBOLO = "Xe"

Case 86 : SIMBOLO = "Rn"



Case Else

SIMBOLO = "NO ES GAS... NOBLE"

End Select

SIMBOLO_ELE = SIMBOLO

End Function

Function PESO_ATOMICO(ByVal ELEC As Integer) As String

Dim PESO As String

Select Case ELEC

Case 2 : PESO = "2.0"

Case 10 : PESO = "20.18"

Case 18 : PESO = "40.0"

Case 36 : PESO = "83.80"

Case 54 : PESO = "131.29"

Case 86 : PESO = "222.0"

Case Else

PESO = "NO ES GAS... NOBLE"

End Select

PESO_ATOMICO = PESO

End Function

End Class



APLICACIÓN 38

1.- ENUNCIADO DEL PROBLEMA:

Desarrollar un programa utilizando funciones que ingresado un valor de pH (entre 0 y 14) devuelva su clasificación (ácido, neutro o base) y un ejemplo de dicha clasificación.

Datos

pH	PAPEL
$0 \leq \text{pH} < 7$ Acido	Rojo
$7 = \text{pH}$ Neutro	No cambia
$7 < \text{pH} \leq 14$ Base	Azúl

pH	Ejemplo
$0 < = \text{pH} < 1$	Solución de Hcl
$1 < = \text{pH} < 2$	Jugo gástrico
$2 < = \text{pH} < 3$	Jugo limón
$3 < = \text{pH} < 4$	Vinagre
$4 < = \text{pH} < 5$	Cerveza
$5 < = \text{pH} < 6$	Te, café
$6 < = \text{pH} < 7$	Leche
$7 = \text{pH}$ Agua pura	Agua pura
$8 \geq \text{pH} > 7$	Agua de mar
$9 \geq \text{pH} > 8$	Levadura
$10 \geq \text{pH} > 9$	Jabón de manos
$11 \geq \text{pH} > 10$	Amoniaco
$12 \geq \text{pH} > 11$	Colisa mineral
$13 \geq \text{pH} > 12$	Hipoclorito de Na
$14 \geq \text{pH} > 13$	Solución de NaOH

Tabla - 22

2.- DISEÑO FORMULARIO-35

Form1

Ingrese el pH

3

CLASIFICAR

BORRAR

Clasificación

ACIDO

Ejemplo

VINAGRE

Color del Papel Tornasol

ROJO

SALIR

3.- CODIFICACION

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

Dim PH As Integer

Dim TIP, D, COL As String

PH = TextBox1.Text

D = CLASIFICACION(PH)

TextBox2.Text = D

TIP = EJEMPLO(PH)

TextBox3.Text = TIP

COL = COLOR_PAPEL(PH)

```
TextBox4.Text = COL
```

```
End Sub
```

```
Function CLASIFICACION(ByVal PH As Integer) As String
```

```
Dim CLAS As String
```

```
Select Case PH
```

```
Case 0, 1, 2, 3, 4, 5, 6 : CLAS = "ACIDO"
```

```
Case 7 : CLAS = "NEUTRA"
```

```
Case 8, 9, 10, 11, 12, 13, 14 : CLAS = "BASE"
```

```
Case Else
```

```
CLAS = "ERROR"
```

```
End Select
```

```
CLASIFICACION = CLAS
```

```
End Function
```

```
Function EJEMPLO(ByVal PH As Integer) As String
```

```
Dim EJEM As String
```

```
Select Case PH
```

```
Case 0 : EJEM = "ACIDO CLORHIDRICO"
```

```
Case 1 : EJEM = "JUGO GÁSTRICO"
```

```
Case 2 : EJEM = "JUGO DE LIMON"
```

```
Case 3 : EJEM = "VINAGRE"
```

```
Case 4 : EJEM = "GASEOSA"
```

```
Case 5 : EJEM = "AGUA DE LLUVIA"
```

```
Case 6 : EJEM = "LECHE"
```

```
Case 7 : EJEM = "AGUA PURA"
```



Case 8 : EJEM = "AGUA DE MAR"

Case 9 : EJEM = "LEVADURA"

Case 10 : EJEM = "JABON DE MANOS"

Case 11 : EJEM = "AMONIACO"

Case 12 : EJEM = "CALIZA MINERAL"

Case 13 : EJEM = "DRANO"

Case 14 : EJEM = "HIDROXIDO DE SODIO"

Case Else

EJEM = "ERROR"

End Select

EJEMPLO = EJEM

End Function

Function COLOR_PAPEL(ByVal PH As Integer) As String

Dim COLOR As String

Select Case PH

Case 0, 1, 2, 3, 4, 5, 6 : COLOR = "ROJO"

Case 7 : COLOR = "NO CAMBIA"

Case 8, 9, 10, 11, 12, 13, 14 : COLOR = "AZUL"

Case Else

COLOR = "ERROR"

End Select

COLOR_PAPEL = COLOR

End Function

End Class



CAPITULO VII

7.1. PROBLEMAS PROPUESTOS A LA INGENIERIA QUIMICA

En este Capítulo se propone una variedad de problemas aplicados a la Ingeniería Química, utilizando principalmente las sentencias de Control como: If then Else / Select Case / do loop / for next / vectores / matrices / procedimientos y funciones de diferentes niveles de dificultad.



PROBLEMAS 01

1.- ENUNCIADO DEL PROBLEMA:

Realice un programa que pida el tamaño de la partícula dispersa en una mezcla y determine qué tipo de mezcla es: solución, coloide o suspensión.

TIPO	TAMAÑO DE LA PARTICULA (Φ)
SOLUCIÓN	$\Phi \leq 1 \text{ nm}$
COLOIDE	$1 \text{ nm} < \Phi \leq 1000 \text{ nm}$
SUSPENSIÓN	$\Phi > 1000 \text{ nm}$

Tabla - 23

T= variable numérico tipo entero, define tamaño de la partícula

2.- DISEÑO FORMULARIO-36

The image shows a screenshot of a Windows application window titled "Form1". The window has a standard Windows title bar with minimize, maximize, and close buttons. The main area of the form is filled with a grid of small dots. On the left side, there is a label that reads "TAMAÑO DE LA PARTICULA". To the right of this label is a text box with the text "TextBox1" inside it. Below the text box, there is a solid black rectangular box. The overall appearance is that of a simple graphical user interface for data entry.

PROBLEMAS 02

1. ENUNCIADO DEL PROBLEMA:

Crear un programa en visual Basic donde ingresando un indicador de longitud de onda que de cómo resultado el color del espectro visible.

Rango de longitud de onda	Color del espectro
380 - 450 nm	Violeta
450 - 495 nm	Azul
495 - 570 nm	Verde
570 - 590 nm	Amarillo
590 - 620 nm	Anaranjado
620 - 750 nm	Rojo

Tabla - 24

2. DISEÑO FORMULARIO-37:

The image shows a screenshot of a Visual Basic form titled "Form1". The form has a background with a black and white wave pattern. A horizontal line divides the form into two sections. The top section is labeled "LONGITUD DE ONDA" and contains a text box for input. Below this text box is a button labeled "CALCULAR". The bottom section is labeled "COLOR" and contains a text box for input. Below this text box is a button labeled "SALIR". In the center of the form, between the two sections, there is a button labeled "BORRAR". The form also has standard Windows window controls (minimize, maximize, close) in the top right corner.

PROBLEMAS 03

1.- ENUNCIADO DEL PROBLEMA:

En un laboratorio se necesita hallar el volumen de un gas con respecto a su temperatura en grados Celsius y la cantidad de moles, utilizando la ecuación general de los gases ideales $PV=RTn$, calcular el volumen en cada caso y convertir de grados Celsius a kelvin.

Donde P: presión en atm, $R= 0.082\text{atm}\cdot\text{l}/\text{k}\cdot\text{mol}$, V= volumen en litros, n=moles del gas. Siguiendo las siguientes condiciones:

$P= 1.2\text{atm}$.

Si la temperatura es entre 50K y 300 K la cantidad de moles es 5

Si la temperatura es entre 300K y 400K la cantidad de moles es 4

Si la temperatura es mayor 400K la cantidad de moles es 2.5

DATOS:

INTERVALOS DE TEMPERATURA $T^{\circ}(\text{K})$	CANTIDAD DE MOLES n
50K – 300k	5
300K – 400K	4
Mayor a 400K	2,5

Tabla - 25

N = nº de moles del gas ideal

P = 1.2atm

T = temperatura

R = constante de los gases ideales: $0.082\text{atm}\cdot\text{l}/\text{k}\cdot\text{mol}$,

V = volumen

$$PV = RTn \longrightarrow V = (0.082/1.2)\cdot T\cdot n$$



Conversión de °C a °K \longrightarrow $^{\circ}\text{K} = ^{\circ}\text{C} + 273.15$

2.- DISEÑO FORMULARIO-38:

The image shows a screenshot of a Windows application window titled "Form1". The window has a standard Windows title bar with minimize, maximize, and close buttons. The background of the form is a dark, grainy image of a landscape with a hill. The form contains the following elements:

- A label "Temperatura EN °C" followed by an empty text input field.
- Two buttons: "Calcular" and "Borrar".
- A label "Volumen (lit)" followed by an empty text input field.
- A label "Cantidad de moles" followed by an empty text input field.
- A label "Temperatura en °K" followed by an empty text input field.
- A "Salir" button at the bottom right.

PROBLEMAS 04

1.- ENUNCIADO DEL PROBLEMA:

Para el programa elaborado se ingresaran datos como la series y nos dará la longitud de onda, utilizando la ecuación de rydberg:

LINEAS	SERIES ESPECTRALES
1	Lyman
2	Balmer
3	Pashen
4	Brackett
5	Pfund

$$\frac{1}{\lambda} = R_h \left(\frac{1}{n_1^2} - \frac{1}{n_2^2} \right)$$

$$R_h = 109678 \text{ cm}^{-1}$$

L = longitud de onda

Tabla – 26

DATOS:

Longi= variable numérica tipo real, longitud de onda.

R= 109678

2.- DISEÑO FORMULARIO-39:

The screenshot shows a window titled "Form5" with standard Windows window controls (minimize, maximize, close). The form contains three input fields and three buttons:

- "Series de:" with a dropdown menu showing "paschen".
- "Linea" with a text input field containing the number "3".
- "Logitud de Onda" with a text input field containing the value "0,00010941118547".
- Buttons labeled "Calcular", "borrar", and "salir" are positioned to the right of the input fields.

Handwritten signature or initials in the bottom left corner.

PROBLEMAS 05

1.- ENUNCIADO DEL PROBLEMA:

En el programa siguiente se inserta una determinada frecuencia y se obtiene la luz de onda que nos resulta si es visible o no al ojo humano y nos da el color asignado para esa longitud de onda

DATOS:

LONGITUD DE ONDA	COLOR ASIGNADO
$3900 \leq \lambda < 4500$	Violeta
$4500 \leq \lambda < 4900$	Azul
$4900 \leq \lambda < 5700$	Verde
$5700 \leq \lambda < 5900$	Amarillo
$5900 \leq \lambda < 6200$	Naranja
$6200 \leq \lambda \leq 7000$	Rojo
$\lambda \geq 7000$	No existe

Tabla - 27

Donde:

$$F = C_i \times a^b \quad l = c / f$$

$$C = 3 \times 10^{10} \quad \lambda = l / (10^{-8})$$

λ = variable numérica tipo real, que representa la longitud de onda.



2.- DISEÑO FORMULARIO-40:

The image shows a screenshot of a Windows application window titled "Form3". The window contains a form with the title "Espectro de luz Visible" in a decorative font. The form has several input fields and buttons. The input fields contain the following text: "7.5", "10", "14", "4000", "Si", and "Violeta". The buttons are labeled "Calcular", "Borrar", and "Salir". The form is enclosed in a decorative border.

Field Label	Value
Wavelength 1	7.5
Wavelength 2	10
Wavelength 3	14
Wavelength 4	4000
Unit	Si
Color	Violeta

Buttons: Calcular, Borrar, Salir

PROBLEMAS 06

1.- ENUNCIADO DEL PROBLEMA:

Realizar un programa, dado el número atómico nos indique su grupo y denominación.

Datos

Z= variable numerica entera, que define el número atomico

NUMERO ATOMICO	GRUPO	DENOMINACION
1,3,11,19,37,55,87	I A	Metales alcalinos
4,12,20,38,56,88	II A	Metales alcalino Téreos
5,13,31,49,81	III A	Téreos
6,14,32,50,82	IV A	Carbonoides
7,15,33,51,83	V A	Nitrogenoides
8,16,34,52,84	VI A	Anfigenos
9,17,35,53,85	VII A	Halogenos
2,10,18,36,54,86	VIII A	Gases nobles

Tabla - 28

2.- DISEÑO FORMULARIO-41:

form1

Ingrese el numero atomico del elemento

Calcular Borrar Salir

PROBLEMAS 07

1.- ENUNCIADO DEL PROBLEMA:

Desarrollar un programa en visual basic que permita calcular la cantidad de volumen a utilizar en una disolución de cualquier ácido fuerte (clorhídrico, bromhídrico, sulfúrico, iorhídrico, nítrico, y perclórico), introduciendo los datos de la densidad, % de pureza, su masa molecular, la molaridad deseada, y el volumen deseado.

DATOS:

Estas son las variables a utilizar:

DEN = densidad

PUR = % de pureza

MM = masa molecular.

MD = molaridad deseada

VD = volumen deseado

C1 = concentración

MASA MOLECULAR	NOMBRE DE ACIDO
36.6	Acido Clorhídrico
98	Acido Sulfúrico
63	Acido Nítrico
128	Acido Iorhidrico
100.5	Acido Perclórico
81	Acido Bromhídrico

Tabla - 29

$$\text{Concentración} = (\text{DE}/0.001) * (\text{PUR}/100) * 1/\text{MM}$$

$$\text{Volumen a utilizar} = (\text{MD} * \text{VD}) / \text{CONCENTRACION}$$



2.- DISEÑO FORMULARIO-42:

CALCULO DE VOLÚMEN PARA ACIDOS FUERTES

DENSIDAD	<input type="text"/>	CALCULAR
% DE PUREZA	<input type="text"/>	
MASA MOLECULAR	<input type="text"/>	BORRAR
MOLARIDAD DESEADA	<input type="text"/>	
VOLUMEN DESEADO (L)	<input type="text"/>	ACEPTAR
	<input type="text"/>	

NOMBRE DEL ACIDO

VOLUMEN A UTILIZAR

SALIR

PROBLEMAS 08

1.- ENUNCIADO DEL PROBLEMA:

Determinar el estado del agua de acuerdo a su temperatura.

TEMPERATURA (°C)	ESTADO DEL AGUA
$T \leq -273.15^\circ$	Improbable
$-273.14^\circ < T < 0.001^\circ$	Solido
$T = 0^\circ$	Solido – liquido
$0.001^\circ < T < 99.99^\circ$	Liquido
$T = 100^\circ$	Liquido – gaseoso
$T > 100.001^\circ$	Gaseoso

Tabla - 30

2.- DISEÑO FORMULARIO-43:

Form1

ESTADOS DEL AGUA A 1 ATM

TEMPERATURA °C

ESTADO

SOLIDO **LIQUIDO** **GASEOSO**

PROBLEMAS 09

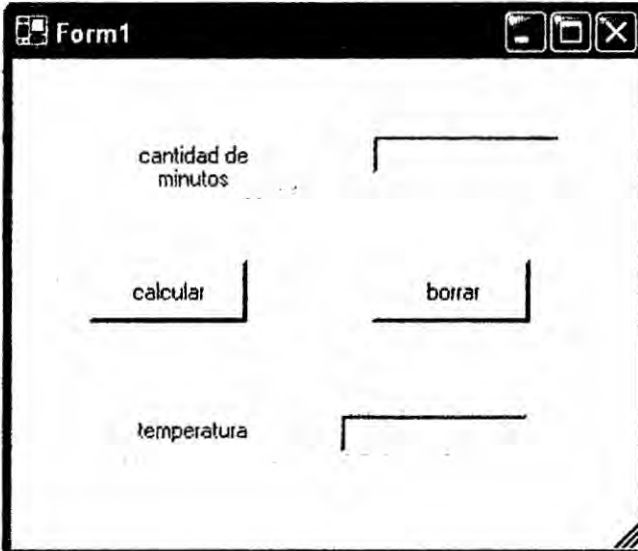
1.- ENUNCIADO DEL PROBLEMA:

Calentamos en laboratorio KSCN en un vaso precipitado, observamos que después de 1min la temperatura aumenta en 1grado, después de 2min aumenta en 2°, después de 3min aumenta en 6°, después de 4min aumenta en 24°, y así sucesivamente. Realizar un programa donde muestre la cantidad de grados que aumenta de acuerdo al número de minutos que se dan.

DATO:

N: cantidad de minutos

2.- DISEÑO FORMULARIO-44:



The image shows a screenshot of a Windows application window titled "Form1". The window contains a form with the following elements:

- A label "cantidad de minutos" followed by a text input field.
- A button labeled "calcular" positioned below the input field.
- A button labeled "borrar" positioned to the right of the "calcular" button.
- A label "temperatura" followed by a text input field.

PROBLEMAS 10

1.- ENUNCIADO DEL PROBLEMA:

Realizar un programa que diseñe aleatoriamente la escala del pH a una temperatura de 25 °C y a 1 atm de presión y muestre por cada valor tantos "*" como indica el valor del pH

$$0 \leq \text{pH} \leq 14$$

2.- DISEÑO FORMULARIO-45:

The image shows a screenshot of a Windows application window titled "Form1". The window contains four buttons: "generar" (top left), "borrar" (top right), "salir" (bottom right), and a large dashed rectangular box in the center. Above the dashed box, the text "pH" is positioned on the left and "histograma" is positioned on the right.

[Handwritten signature]

PROBLEMAS 11

1.- ENUNCIADO DEL PROBLEMA:

Diseñe un programa en Visual Basic que genere 10 números aleatorios en el intervalo de 0 a 14, cada número determina el PH o el POH de una solución, determinar:

- El número de soluciones acidas
- El número de soluciones neutras
- El número de soluciones básicas

Fórmula para calcular numero aleatorios enteros

Datos

$\text{INT} ((\text{lim sup} - \text{lim inf} + 1) \times \text{rnd} ())$

$\text{PH} = \text{int} (14 - 0 + 1 \times \text{rnd}() + 0)$

$\text{NRO}(I) = \text{PH O POH de la solución}$

2.- DISEÑO FORMULARIO-46:

The screenshot shows a Visual Basic form window titled "Form1". It contains four buttons: "GENERAR", "CALCULAR", "BORRAR", and "SALIR". Below the buttons, there are three rows of data, each with a list of numbers on the left, a label in the middle, and an input field on the right. The first row shows numbers 8, 10, 12, 11, 1, 8, 12, 6, 9, 7 and the label "#SOL. ACIDAS" with an input field containing the value 2. The second row shows the same numbers and the label "#SOL. NEUTRAS" with an input field containing the value 1. The third row shows the same numbers and the label "#SOL. BASICAS" with an input field containing the value 7.

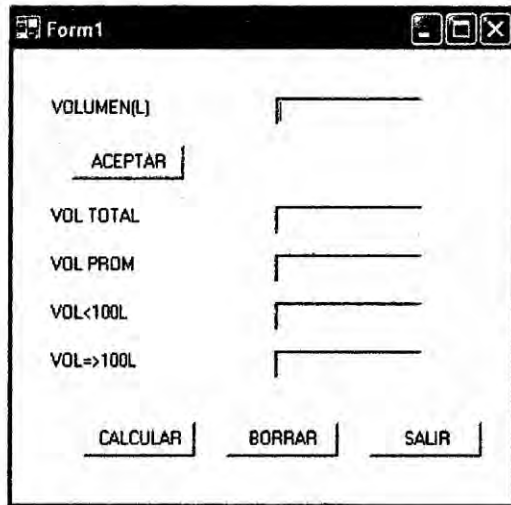
Numbers	Label	Value
8 10 12 11 1 8 12 6 9 7	#SOL. ACIDAS	2
8 10 12 11 1 8 12 6 9 7	#SOL. NEUTRAS	1
8 10 12 11 1 8 12 6 9 7	#SOL. BASICAS	7

PROBLEMAS 12

1.- ENUNCIADO DEL PROBLEMA:

En la empresa SOL GAS, se desea saber el volumen ingresado a la planta, la cual está dividida en balones mayores y menores de 100 L y el volumen total y promedio ingresado.

2.- DISEÑO FORMULARIO-47:



The image shows a screenshot of a Windows application window titled "Form1". The window contains the following elements:

- A label "VOLUMEN(L)" followed by a text input field.
- A button labeled "ACEPTAR" below the first input field.
- A label "VOL TOTAL" followed by a text input field.
- A label "VOL PROM" followed by a text input field.
- A label "VOL < 100L" followed by a text input field.
- A label "VOL > 100L" followed by a text input field.
- At the bottom, three buttons: "CALCULAR", "BORRAR", and "SALIR".



PROBLEMAS 13

1.- ENUNCIADO DEL PROBLEMA:

Realizar un programa que genere 3 numeros aleatorios indicando el número de orden los elemento y su respectivo nombre consernientes al grupo de los gases nobles de la tabla periodica, asi como se muestra en la siguiente tabla:

Nº DE ORDEN	SIMBOLO	ELEMENTO
1	He	Helio
2	Ne	Neón
3	Ar	Argón
4	Kr	Kriptón
5	Xe	Xenón
6	Rn	Radón

Tabla - 31

2.- DISEÑO FORMULARIO-48:

The screenshot shows a Windows form titled "Form1" with a standard title bar (minimize, maximize, close buttons). The form contains three buttons: "Generar" at the top left, "Borrar" at the top right, and "Salir" at the bottom right. In the center of the form is a list box containing the following text:

```
5 Xe(XeNon)
1 He(Heño)
1 He(Heño)
```

PROBLEMAS 14

1.- ENUNCIADO DEL PROBLEMA:

Diseñe un programa que calcule la presión de un gas ideal, a diferentes tipos de volumen y temperatura. Además debe registrar la presión máxima y mínima del gas, con su respectiva temperatura y su respectivo volumen, así como se muestra en la siguiente tabla: /

	VOL1	VOL2	VOL3
TEMP1	PRES1,1	PRES1,2	PRES1,3
TEMP2	PRES2,1	PRES2,2	PRES2,3
TEMP3	PRES3,1	PRES3,2	PRES3,3

Tabla - 32

Datos:

N: número de moles del gas

VOL j: variable tipo real que representa el volumen en litros

TEMP i: variable tipo real que indica la temperatura en grados Celsius

Incógnita:

PRES i, j: representa la presión en atmósferas

$P = 0,0082 \times (\text{temp (f)} + 273) \times n/\text{vol (c)}$

PROBLEMAS 15

1.- ENUNCIADO DEL PROBLEMA:

Programa que calcula la presión ejercida por N moles de un GAS REAL, que ocupa un volumen V (litros) a una temperatura T en K , incluye las constantes críticas.

La ecuación de Van Der Waals: $P = \frac{N \times R \times T + a \times b / V^2}{V - b}$

R= 0,082

2.- DISEÑO FORMULARIO-49:

The image shows a screenshot of a graphical user interface (GUI) for a program. The window title is "Form1". The interface contains several input fields and buttons. The input fields are labeled: "Volumen del gas Real", "Temperatura", "Volumen", "Numero de moles", "conste. critica a", and "conste. critica b". There is also an output field labeled "Presion". At the bottom of the window, there are three buttons: "CALCULAR", "BORRAR", and "SALIR".

PROBLEMAS 16

1.- ENUNCIADO DEL PROBLEMA:

En una planta química se observa que un determinado gas logra desplazar a una cierta distancia el embolo que lo presiona. Se desea saber cual es el trabajo efectuado por dicho gas cuando este desplaza al embolo y averiguar cuántos equivalentes gramos hay en la variación de volumen .Para tales cálculos se pide realizar un programa en visual Basic que me permita hallar lo pedido.

Datos:

N = Normalidad

M = Molaridad

P = Presión

W = Trabajo

V = Volumen que se desplaza el embolo

Numero Eq – g = K = N * V

Trabajo = W = P * V * 24.218

2.- DISEÑO FORMULARIO-50:

The screenshot shows a Visual Basic form titled "Form1" with the following elements:

- Input field for "PRESION" with the value "5" and unit "atm".
- Input field for "VOLUMEN" with the value "6" and unit "litros".
- Input field for "normalidad" with the value "0,2" and unit "N".
- Buttons "CALCULAR" and "BORRAR".
- Output field for "TRABAJO EFECTUADO POR EL GAS" with the value "726,54" and unit "cal".
- Output field for "NUMERO DE Eq_g" with the value "1,2".
- A "SALIR" button at the bottom.

PROBLEMAS 17

1.- ENUNCIADO DEL PROBLEMA:

Programa que calcula la presión ejercida por N moles de un gas, que ocupa un volumen V (litros) a una temperatura T(K) ; la presión dada en atmósferas.

Donde el gas se ajusta a la ecuación de gases ideales:

$$P = \frac{nRT}{V}$$

$$R = 0,082 \text{ atm.L} \\ \text{mol K}$$

2.- DISEÑO FORMULARIO-51:

VOLUMEN DE GASES IDEALES

Temperatura

Volumen

Moles

CALCULAR BORRAR

Presion

SALIR

PROBLEMAS 18

1.- ENUNCIADO DEL PROBLEMA:

Programa que calcula la capacidad calorífica (C_p) en $\text{cal}/^\circ\text{K}\cdot\text{mol}$ del hierro a la temperatura T en Kelvin sabiendo que:

Estado	Capacidad calorífica a presión constante	Rango de temperatura $^\circ\text{K}$
Cristalino α	$4,26 + 0,00640 * T$	273 - 626
Cristalino β	$6,99 + 0,000905 * T$	626 - 1725
Líquido	8,55	1725 - 1903

Tabla - 33

Datos:

T = variable numérica tipo real, define temperatura en grados $^\circ\text{k}$

CAPACIDAD_CALORIFICA: nombre de la función.

CP: variable numérico real, resultado de la capacidad calorífica.

2.- DISEÑO FORMULARIO-52:

Form1

Capacidad calorifica del Hierro

Temperatura

Capacidad calorifica

Estado del Hierro

Rango de temperatura

PROBLEMAS 19

1.- ENUNCIADO DEL PROBLEMA:

Desarrollar una Función en la que se ingresa la temperatura caliente y temperatura fría nos dé como respuesta la eficiencia según el ciclo de Carnot.

DATOS

TC= variable numérica real, que define la temperatura caliente

TF= variable numérica real, que define la temperatura fría

EF variable numérica real, que define la eficiencia

$$EF = (TC - TF) / TC$$

2.- DISEÑO FORMULARIO-53:

Form1

EFICIENCIA DE UN CICLO DE CARNOT

INGRESE TEMPERATURA CALIENTE

INGRESE TEMPERATURA FRÍA

ACEPTAR BORRAR

EFICIENCIA

SALIR

BIBLIOGRAFIA

1. <http://www.canalvisualbasic.net/>
2. [http://msdn.microsoft.com/es-es/library/32s6akha\(VS.80\).aspx](http://msdn.microsoft.com/es-es/library/32s6akha(VS.80).aspx)
3. [http://msdn.microsoft.com/es-es/library/y6yz79c3\(VS.80\).aspx](http://msdn.microsoft.com/es-es/library/y6yz79c3(VS.80).aspx)
4. <http://users.dsic.upv.es/~ingarcia/Castellano/Docencia/FI2/Descarga/T05.pdf>
5. [http://msdn.microsoft.com/es-es/library/wak0wfyf\(VS.80\).aspx](http://msdn.microsoft.com/es-es/library/wak0wfyf(VS.80).aspx)
6. Cristian Sánchez Flores Desarrollando aplicaciones con
Visual Basic 2008
Editora Macro,
1ra, Edición octubre 2008-Peru
7. Francisco Charre Ojeda Programación Visual Basic 2008
Editora Anaya
2009, Madrid
8. Evangelos Petroustos La Biblia
Visual Basic 2008
Editora Anaya- 2009
9. Peter Norton Introducción a la Computación
Mc Graw-Hill
Interamericana, 2da. Edición, México.
10. Osvaldo Cairo Battistutti Metodología de la Programación tomo I y II
(Algoritmos, diagramas de flujo y programas)
1995



11. Eduardo Alcalde-M. García Metodología de la Programación.

12. Julio Vásquez Paragulla Diseño de Programación
(Pseudocódigo, Diagrama de Flujo)
3era Edición – 1997

