



## Documento de Planeación didáctica

PARTE GENERAL	
<b>NOMBRE DEL PROFESOR</b>	Juan Gutiérrez Quiroz
<b>SUBSISTEMA Y NIVEL ACADÉMICO</b>	Colegio de Ciencias y Humanidades, Plantel Oriente.
<b>ASIGNATURA / SEMESTRE O AÑO</b>	CIBERNÉTICA Y COMPUTACIÓN II. Sexto Semestre.
<b>UNIDAD TEMÁTICA Y CONTENIDOS</b>	<b>UNIDAD III. PROCEDIMIENTOS</b> La modularidad como una herramienta para el manejo de programas <ul style="list-style-type: none"><li>○ Concepto.</li><li>○ Importancia.</li><li>● Procedimientos<ul style="list-style-type: none"><li>○ Concepto de procedimiento.</li><li>○ Definición, declaración e invocación de procedimientos</li></ul></li></ul> Alcance de identificadores. <ul style="list-style-type: none"><li>○ Variables globales y locales.</li><li>○ Nombre de procedimientos</li></ul>
<b>OBJETIVOS DE LA UNIDAD</b>	Al finalizar la unidad, el alumno utilizará los procedimientos, para elaborar programas de estructura modular, mediante el desarrollo de programas de solución de problemas.
<b>DURACIÓN</b>	4 HORAS, 2 SESIONES DE 2 HORAS CADA UNA
<b>POBLACIÓN</b>	20 alumnos.
<b>BIBLIOGRAFÍA</b>	<ul style="list-style-type: none"><li>● Avila, R. Juventino. Unidad 2 Procedimientos y Funciones. Recuperado el 9 de marzo de 2017, de: <a href="http://www.cch-vallejo.unam.mx/tareas/Files/Func&amp;Procs.doc">http://www.cch-vallejo.unam.mx/tareas/Files/Func&amp;Procs.doc</a></li></ul>



- Santos, F. Benito, Desarrollo de una aplicación basada en Java para la enseñanza de la programación en Pascal. (2001). Recuperado el 9 de marzo de 2017 de:  
<http://teleformacion.edu.aytolacoruna.es/PASCAL/document/modular.htm>
- UNAM, CCH. CIBERNÉTICA Y COMPUTACIÓN II (2012). Unidad III Procedimientos y funciones. Recuperado el 9 de marzo de 2017 de  
<http://portalacademico.cch.unam.mx/materiales/prof/matdidac/sitpro/mate/ciber/ciber2/funciones/unidad3p/unidad3p.html>



## Actividad 1. Actividad de inicio

(Esta actividad se realiza para empezar a trabajar una unidad temática)

TÍTULO DE LA ACTIVIDAD	
OBJETIVO DE APRENDIZAJE	<p>Que el alumno:</p> <ul style="list-style-type: none"><li>• Comprenda la importancia de dividir un programa en módulos.</li><li>• Identifique en la estructura de programas, la zona para la declaración e invocación de procedimientos.</li><li>• Describa el concepto de programación modular, procedimientos y tipos de parámetros.</li></ul>
RECURSOS	<ul style="list-style-type: none"><li>• Computadora</li><li>• Lenguaje de programación Free Pascal</li><li>• Conexión a Internet</li><li>• Memoria USB</li><li>• Cañón (Videoprojector)</li><li>• Pantalla</li></ul>
DESCRIPCIÓN DE LAS ACTIVIDADES	
TAREAS EN EL ORDEN EN QUE SE REALIZAN	<p><b>Trabajo previo a la clase (2 horas, aproximadamente)</b></p> <ul style="list-style-type: none"><li>• <b>El alumno, de forma individual,</b> Investiga y desarrolla los conceptos indicados por el profesor. Para ello <i>sugiere consultar los materiales, propuestos en la bibliografía:</i><ul style="list-style-type: none"><li>○ <i>Unidad 2 Procedimientos y Funciones</i></li><li>○ <i>Desarrollo de una aplicación basada en Java para la enseñanza de la programación en Pascal.</i></li><li>○ <i>UNAM, CCH. CIBERNÉTICA Y COMPUTACIÓN II (2012). Unidad III Procedimientos y funciones.</i></li></ul></li></ul> <p><b>El profesor indicará el material en el que se investigue cada uno de los conceptos.</b></p> <p><i>Ver Anexo 1 (el cuestionario del profesor)</i></p> <p><b>SESIÓN 1: 2 HORAS</b></p>



## APERTURA

### Actividad para iniciar la clase.

- **El alumno** entrega un reporte escrito de la investigación, con las características que el profesor indique.
- **El profesor**, en plenaria, propiciará una lluvia de ideas con los alumnos con base a la pregunta ¿Qué entiendes por programación modular?, para que esto sea el inicio de un debate en el que puedan deducir los conceptos básicos la programación por módulos, que se investigaron.
- **El profesor**, en base a una **Participación Guiada** induce al grupo a generar una definición propia de los conceptos que se investigaron.

## DESARROLLO.

### Actividades durante la clase.

- **El profesor**, a manera de recordatorio y en base a la **participación guiada** de los alumnos, indicará los pasos a seguir en el desarrollo de un programa. Después explicará cómo utilizar un procedimiento, en donde se declara, tipos de variables, como se integra un programa que utilice procedimientos. Para que los alumnos refuercen la comprensión de los conceptos, vistos en clase, sugiere consultar los materiales propuestos en la bibliografía.

*Ver Anexo. 2*

- **El profesor** guía a los alumnos para generar la definición de procedure, explica cuándo es conveniente utilizarlo, donde se declara y utiliza un ejemplo sencillo para que los alumnos comprendan y apliquen este concepto.
- **Los alumnos, en equipos de 2 personas**, aplican los conceptos relacionados con el uso de los procedures, realizarán una práctica guiada que consistirá en capturar un programa y después realizarán los cambios indicados por él profesor.

*Ver Anexo 3*

## CIERRE

### Actividad para concluir la clase.

- **El profesor** ejemplifica, utilizando diagramas de flujo y pseudocódigo, como elaborar un programa que utilice varios procedures. *Ver anexo 4*
- **El profesor** promueve *una reflexión*, en grupo, en base a preguntas y



	<p>respuestas con los alumnos.</p> <p><b>Trabajo extraclase para los alumnos. (2 horas, aproximadamente)</b></p> <ul style="list-style-type: none"><li>• <b>El alumno (individual):</b> Desarrollará un algoritmo para un proceso que realice alguna de las cuatro operaciones básicas (suma, resta, multiplicación y división) con dos números enteros. El proceso debe terminar cuando el usuario decida. El alumno debe traer el algoritmo en su cuaderno de apuntes para que pueda participar y proponer ideas para obtener una solución en grupo.</li></ul>
<b>EVIDENCIAS DE APRENDIZAJE DEL ALUMNO</b>	<ul style="list-style-type: none"><li>• El reporte escrito del cuestionario de investigación de forma individual <i>Ver anexo 1.1</i></li></ul>
<b>FORMA DE EVALUACIÓN</b>	<ul style="list-style-type: none"><li>• Reporte de investigación conforme a las lista de cotejo 80%</li><li>• Participación 20% <i>Ver rúbrica en el anexo 7</i></li></ul>



## Actividad 2. Actividad de desarrollo

(Esta actividad se realiza para trabajar a lo largo de una unidad temática)

TÍTULO DE LA ACTIVIDAD	
Elaboración de un programa	
OBJETIVO DE APRENDIZAJE	Que el alumno elabore programas utilizando los conceptos de la programación modular.
RECURSOS	<ul style="list-style-type: none"><li>• Computadora</li><li>• Lenguaje de programación Free Pascal</li><li>• Conexión a Internet</li><li>• Memoria USB</li><li>• Cañón (Videoprojector)</li><li>• Pantalla (Deseable, de otra forma trabajaremos con las condiciones del salón de clase).</li></ul>
DESCRIPCIÓN DE LAS ACTIVIDADES	
TAREAS EN EL ORDEN EN QUE SE REALIZAN	<p><b>SESIÓN 2: 2 HORAS.</b></p> <p><b>APERTURA</b> <b>Actividad para iniciar la clase.</b></p> <ul style="list-style-type: none"><li>• <b>Los alumnos, algunos de ellos, (de manera individual),</b> exponen en el pizarrón, el algoritmo realizado como trabajo extraclase.</li><li>• <b>El profesor</b> modera, y mediante una participación guiada promueve la interacción con el grupo y entre los mismos alumnos, para que a través de La discusión y exploración conjunta de las ideas sea posible obtener el algoritmo adecuado.</li></ul> <p><b>DESARROLLO</b></p> <ul style="list-style-type: none"><li>• <b>El profesor:</b> Con base en el algoritmo obtenido, propone el siguiente ejercicio: “Desarrollar un programa que realice alguna de las cuatro operaciones básicas (suma, resta, multiplicación o división) con dos números enteros”, explica la estructura general del código del programa en Pascal y ejemplifica el desarrollo del procedure para la operación suma. <i>Ver Anexo 5.</i></li><li>• <b>Los alumnos (de forma grupal), en equipos de 2 personas,</b> terminan el desarrollo del programa que realice alguna de las cuatro operaciones</li></ul>



	<p>básicas (suma, resta, multiplicación o división) con dos números enteros. El programa debe terminar cuando el usuario decida.</p> <ul style="list-style-type: none"><li>• <b>El profesor</b> supervisa y resuelve las dudas de los alumnos surgidas al ir desarrollando el programa.</li></ul> <p><b>CIERRE</b></p> <ul style="list-style-type: none"><li>• <b>Los alumnos, por equipos</b>, y con ayuda del cañón, exponen el programa obtenido.</li></ul> <p><b>Actividades extraclase.</b></p> <ul style="list-style-type: none"><li>• <b>El alumno, de forma individual</b>, realiza un reporte escrito, del programa desarrollado, que incluya las etapas de ciclo de desarrollo de programas visto en clase y además muestre la ejecución del mismo. Lo entrega al profesor. <i>Ver anexo 5.1.</i></li><li>• <b>El profesor</b> indica a los alumnos que formen equipos de 2 personas y que propongan un problema que pueda ser desarrollado con las herramientas de programación vistas hasta ahora.</li></ul>
<p><b>EVIDENCIAS DE APRENDIZAJE DEL ALUMNO</b></p>	<p>El reporte escrito entregado por el alumno.</p>
<p><b>FORMA DE EVALUACIÓN</b></p>	<ul style="list-style-type: none"><li>• Reporte del desarrollo del programa que realice alguna de las cuatro operaciones básicas (suma, resta, multiplicación o división) con dos números enteros. El programa termina cuando el usuario decida. <i>Ver anexo 5.1</i></li><li>• El reporte entregado será evaluado conforme a la lista de cotejo 80% <i>Ver Anexo 7</i></li><li>• Participación 20%</li></ul>



### Actividad 3. Actividad de cierre

(Esta actividad se realiza para concluir el trabajo de una unidad temática)

TÍTULO DE LA ACTIVIDAD	
Exposición del proyecto	
<b>OBJETIVO DE APRENDIZAJE</b>	<p>Que el alumno:</p> <ul style="list-style-type: none"><li>• Comprenda la utilidad de manejar parámetros globales como medios de comunicación de información entre los módulos y el programa.</li><li>• Explique las diferencias entre los parámetros por valor, referencia y variable.</li></ul>
<b>RECURSOS</b>	<ul style="list-style-type: none"><li>• Computadora</li><li>• Lenguaje de programación Free Pascal</li><li>• Conexión a Internet</li><li>• Memoria USB</li><li>• Cañón (Videoprojector)</li><li>• Pantalla</li></ul>
<b>DESCRIPCIÓN DE LAS ACTIVIDADES</b>	
<b>TAREAS EN EL ORDEN EN QUE SE REALIZAN</b>	<p><b>SESIÓN 3: 2 HORAS.</b></p> <p><b>APERTURA</b></p> <ul style="list-style-type: none"><li>• <b>El profesor</b> formaliza los equipos de 2 personas y el programa que propusieron. Indica los elementos que debe incluir la presentación que utilizarán para la exposición del proyecto y que posteriormente van a entregar.</li></ul> <p><b>DESARROLLO</b></p> <ul style="list-style-type: none"><li>• <b>Los alumnos en equipos de 2 personas</b>, plantean la solución del problema e inician el desarrollo del programa.</li></ul> <p><b>CIERRE</b></p> <ul style="list-style-type: none"><li>• <b>Los alumnos en equipo</b>, preparan la presentación para la exposición del proyecto la siguiente clase.</li></ul> <p><b>Actividad extraclase</b></p> <ul style="list-style-type: none"><li>• <b>Los alumnos, en equipos de dos personas</b>, terminan el programa y la presentación para exponer frente al grupo.</li></ul>



### Actividad 3. Actividad de cierre Sesión 4 (se aumentó) (Exposición del proyecto)

TÍTULO DE LA ACTIVIDAD	
OBJETIVO DE APRENDIZAJE	Exposición del proyecto  Que el alumno: <ul style="list-style-type: none"><li>• Presente ante el grupo una solución a un problema planteado por el mismo.</li></ul>
RECURSOS	<ul style="list-style-type: none"><li>• Computadora</li><li>• Lenguaje de programación Free Pascal</li><li>• Conexión a Internet</li><li>• Memoria USB</li><li>• Cañón (Videoprojector)</li><li>• Pantalla</li></ul>
DESCRIPCIÓN DE LAS ACTIVIDADES	
TAREAS EN EL ORDEN EN QUE SE REALIZAN	<b>SESIÓN 4: 2 HORAS.</b> <b>APERTURA</b> <ul style="list-style-type: none"><li>• <b>El profesor</b> indica la forma en que se han de llevar a cabo las presentaciones.</li></ul> <b>DESARROLLO</b> <ul style="list-style-type: none"><li>• <b>Los alumnos en equipos de 2 personas</b>, con ayuda del cañón, exponen el desarrollo de programas que solucionan los problemas propuestos por ellos mismos.</li></ul> <b>CIERRE</b> <ul style="list-style-type: none"><li>• <b>Comentarios y conclusiones.</b></li></ul>
EVIDENCIAS DE APRENDIZAJE DEL ALUMNO	Presentación elaborada por los equipos. <i>Ver anexo 6</i>
FORMA DE EVALUACIÓN	Para esta actividad. Presentación conforme a las lista de cotejo 20% Exposición 80% <i>Ver anexo 7</i>



## Anexos

ANEXO	DESCRIPCIÓN	PÁGINA
Anexo 1	El cuestionario.	11
Anexo 1.1	Reporte de investigación por el alumno	12
Anexo 2	Ciclo de desarrollo de un programa	13
Anexo 3	Definición de procedure y ejemplo	14
Anexo 4	Ejemplo programa modular	15
Anexo 5	Ejemplo explicación del profesor	16
Anexo 5.1	Ejemplo desarrollado por los alumnos	18
Anexo 6	Evidencias del proyecto	19
Anexo 7	Lista de cotejo	20



## ANEXO 1

### Conceptos de programación modular (cuestionario)

#### ¿Qué es programación modular?

Uno de los métodos más conocidos para resolver un problema es dividirlo en problemas más pequeños, llamados subproblemas. De esta manera, en lugar de resolver una tarea compleja y tediosa, resolvemos otras más sencillas y a partir de ellas llegamos a la solución. Esta técnica se usa mucho en programación ya que programar no es más que resolver problemas, y se le suele llamar diseño descendente, metodología del divide y vencerás o programación Top-Down. Es evidente que si esta metodología nos lleva a tratar con subproblemas, entonces también tengamos la necesidad de poder crear y trabajar con subprogramas para resolverlos. A estos subprogramas se les suele llamar módulos, de ahí viene el nombre de programación modular. En Pascal disponemos de dos tipos de módulos: los procedimientos y las funciones.

#### Método de Programación Top Down

En el modelo Top Down se formula un resumen del sistema, sin especificar detalles. Cada parte del sistema se refina diseñando con mayor detalle. Cada parte nueva es entonces redefinida, cada vez con mayor detalle, hasta que la especificación completa es lo suficientemente detallada para validar el modelo. El modelo "Top Down" se diseña con frecuencia con la ayuda de "cajas negras" que hacen más fácil cumplir requerimientos aunque estas cajas negras no expliquen en detalle los componentes individuales.

#### ¿Qué es una subrutina en Pascal?

Una subrutina es una porción de código que forma parte de un programa más grande. Esa porción de código realiza una tarea específica, relativamente independiente del resto del código. La mayoría de los lenguajes de programación soportan la creación de subrutinas y métodos para llamarlas (invocarlas) y retornarlas.

Las subrutinas en Pascal son, partes separadas de código que ejecutan tareas pequeñas de un programa grande.

#### ¿Qué es un Procedure y cuándo se utiliza?

Es un programa diseñado en forma exclusiva para algo específico que se desee realizar.

Se incluye después de la declaración de variables y cuando se requiere realizar la acción para la cual fue diseñado es ejecutado desde el programa principal.

#### ¿Qué es una función y cuándo se utiliza?

Es un grupo de instrucciones con un objetivo en particular y que se ejecuta al ser llamada desde otra función o procedimiento. Una función puede llamarse múltiples veces e incluso llamarse a sí misma (función recurrente).

Las funciones pueden recibir datos desde afuera al ser llamadas a través de los parámetros y deben entregar un resultado.

**\*\* Para esta investigación el profesor sugiere consultar los materiales:**



- *Unidad 2 Procedimientos y Funciones*
- *Desarrollo de una aplicación basada en Java para la enseñanza de la programación en Pascal.*
- *UNAM, CCH. CIBERNÉTICA Y COMPUTACIÓN II (2012). Unidad III Procedimientos y funciones.*

***enunciados en la sección de recursos, pero el alumno puede consultar otro tipo de materiales si así lo desea.***

## ANEXO 1.1

**Reporte del cuestionario entregado por 2 alumnos.**

**Se pueden consultar en:**

### **Reporte 1**

<https://drive.google.com/file/d/0B1CZTV2fLaf8WFIONmNhT1R5ZXM/view?usp=sharing>

### **Reporte 2**

<https://drive.google.com/file/d/0B1CZTV2fLaf8OXMzMFIKRkF3SWc/view?usp=sharing>



## ANEXO 2

### **Pasos a seguir para el desarrollo de un programa (Recordatorio)**

Pasos a seguir para el desarrollo de un programa utilizando la programación modular.

Recordemos los pasos a seguir en el desarrollo de un programa:

#### ***Entender el problema.***

Aquí analizamos el problema y visualizamos las posibles soluciones, las variables de entrada y las salidas.

#### ***Desarrollo del Algoritmo.***

Desarrollaremos un algoritmo aplicando el método Top-Down, esto es desarrollaremos en primera instancia el programa principal y después cada uno de los módulos requeridos.

#### ***Prueba de escritorio.***

De acuerdo al algoritmo obtenido realizaremos las pruebas necesarias, con datos de entrada, para comprobar que obtenemos los resultados esperados.

#### ***Codificación***

Después de verificar el algoritmo procederemos a desarrollar el código del programa.

#### ***Ejecución.***

Cuando terminemos de codificar nuestro programa, realizaremos las depuraciones correspondientes: corregir posibles errores de sintaxis, y corrección de errores lógicos, además de la muestra de resultados.

Con lo anterior concluimos el desarrollo del programa, finalmente se realizará una exposición frente a grupo.



## Anexo 3

### Definición de procedure y ejemplo

Un Procedure es un programa diseñado para realizar una tarea específica, esta tarea es realizada cuando el procedure es llamado desde el programa principal.

Los procedures se declaran, en un programa en Pascal, después de la sección de variables, a continuación ejemplificamos la creación de un procedure en su forma más sencilla<sup>1</sup>.

```
PROGRAM Procedimiento;
```

```
PROCEDURE Titulo;
```

```
BEGIN
```

```
Writeln('Programa de Turbo pascal');
```

```
END;
```

```
BEGIN
```

```
Writeln ('Programa ejemplo del uso de procedimientos');
```

```
Titulo;
```

```
{Llama al procedimiento}
```

```
WriteLn;
```

```
Titulo;
```

```
{Vuelve a llamar al procedimiento}
```

```
END.
```

El programa anterior lo que hace es escribir el título "Programa de Turbo pascal", para esta función específica se creó el procedure Título, como no se necesitan variables, es lo primero que se escribe al codificar el programa, después, en el programa principal, cada vez que queramos escribir el título, se llama la procedure *Título*.

---

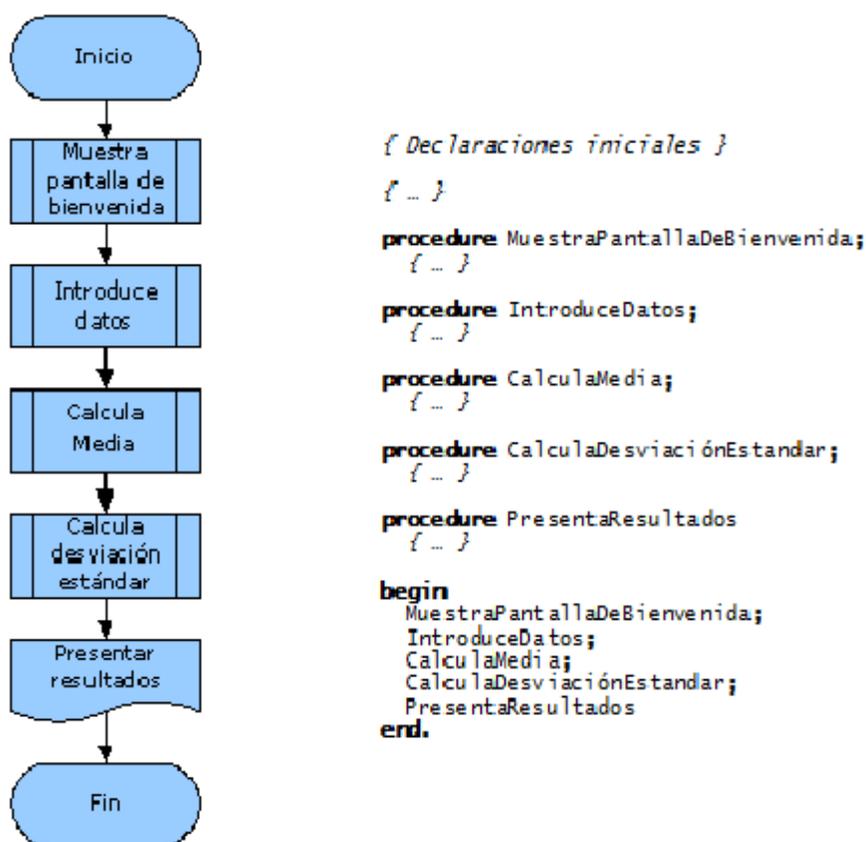
<sup>1</sup> UNAM, CCH. CIBERNÉTICA Y COMPUTACIÓN II (2012). Unidad III Procedimientos y funciones. Recuperado el 9 de marzo de 2017 de <http://portalacademico.cch.unam.mx/materiales/prof/matdidac/sitpro/mate/ciber/ciber2/funciones/unidad3p/unidad3p.html>



## Anexo 4

### Programa modular

Ejemplo: Dibujar el diagrama de flujo y escribir los pasos para el cálculo de la desviación estándar de un conjunto cualquiera de números.



El símbolo de los rectángulos significa proceso predefinido. Por otro lado se puede visualizar que los procedimientos primero se escriben y después son llamados en el programa principal.

Ejemplo tomado de Avila, R. Juventino. Unidad 2 Procedimientos y Funciones. Recuperado el 9 de marzo de 2017, de: <http://www.cch-vallejo.unam.mx/tareas/Files/Func&Procs.doc>



## Anexo 5

### Explicación del profesor

### Desarrollo del programa

Desarrollo del algoritmo para elaborar un programa que realice alguna de las cuatro operaciones básicas (suma, resta, multiplicación o división) con dos números enteros.

1. Planteamiento del problema.

El usuario debe elegir de un menú de opciones una de las cuatro operaciones básicas:

- *Suma*
- *Resta*
- *Multiplicación*
- *División*
- *Salir.*

2.- El grupo debe llegar un algoritmo como el que se muestra:

Para realizar este algoritmo iremos de lo general a lo particular es decir, de arriba hacia abajo (Top-Down), el algoritmo es el siguiente:

Algoritmo

Ø Inicio

Ø *repetir*

Ø Escribe el menú principal

Ø *suma*

Ø *resta*

Ø *multiplicación*

Ø *división*

Ø *salir*

Ø Escribir *¿qué opción eliges?*

Ø Leer opción

Ø Ejecuta la operación seleccionada

Ø En caso de que la opción sea:

1: suma

2: resta



3: multiplicación

4: división

5: salir

∅ *Hasta que la opción sea igual a 5*

∅ Fin

A continuación el profesor guiará a los alumnos para desarrollar el algoritmo del procedimiento para la suma y los alumnos deben desarrollar los subprogramas correspondientes para la resta, multiplicación y división, después debe integrar todos los módulos e implementar la opción salir. El profesor estará supervisando en todo momento a los alumnos.

**para la operación suma el algoritmo es:**

∅ Procedure suma;  
∅ Var A, B, C: integer;  
∅ Begin  
∅ Leer A y B  
∅ C:=A+B  
∅ Escribir (C)  
∅ End; (\*suma\*)

Estos módulos serán llamados desde el programa principal, el algoritmo y pseudocódigo es el siguiente:

∅ (\*Programa Principal\*)  
∅ *Begin*  
∅ Repite  
∅ Limpiar pantalla  
∅ Escribir ('1: suma');  
∅ Escribir ('2: resta');  
∅ Escribir ('3: Multiplicación');  
∅ Escribir ('4: División');  
∅ Escribir ('5: Salir');  
∅ Escribir (opción);  
∅ Leer (opción)  
∅ *Case* opción *of*  
∅ 1: **suma**; (\* llamado del procedimiento suma\*)  
∅ 2: **resta**;  
∅ 3: **Multiplica**;  
∅ 4: **Divide**;  
∅ 5: Escribir ('Fin del programa');  
∅ *Else*  
∅ Escribir ('Opción invalida');



Ø End; (\*case\*)  
Ø ¿Desea Continuar?  
Ø Until upcase (R)= 'N';  
Ø End.

El siguiente paso es codificar el programa.

## Anexo 5.1

**Ejemplo desarrollado por los alumnos.**

**Los ejemplos se pueden consultar en los siguientes enlaces:**

### **Trabajo 1**

<https://drive.google.com/file/d/0B1CZTV2fLaf8SG05TDRVSVJKRzg/view?usp=sharing>

### **Trabajo 2**

<https://drive.google.com/file/d/0B1CZTV2fLaf8cEIWOXQ0WINSMjQ/view?usp=sharing>



## **Anexo 6**

### **Evidencias del proyecto**

#### **Reporte 1**

<https://drive.google.com/file/d/0B1CZTV2fLaf8Q0VQbFVRVFhHbmc/view?usp=sharing>

#### **Reporte 2**

<https://drive.google.com/file/d/0B1CZTV2fLaf8MkRISkJuMTBKUE0/view?usp=sharing>

#### **Reporte 3**

<https://drive.google.com/file/d/0B1CZTV2fLaf8cGdLb0ZfX2RIMG8/view?usp=sharing>



## ANEXO 7

### LISTA DE COTEJO 1

ELEMENTOS A EVALUAR	VALOR	SI	NO
Tiene portada	20%		
Incluye las ideas principales	40%		
Respeto el orden de la presentación de ideas	20%		
La redacción es coherente	10%		
La presentación es buena	10%		
<b>TOTAL</b>	<b>100%</b>		
<b>OBSERVACIONES:</b> En las actividades se trabaja por equipo, pero la entrega de los reportes es personal, El reporte tiene un valor del 80% por ciento de la evaluación de la actividad y la participación en clase del 20%.			



## CIBERNÉTICA Y COMPUTACIÓN II

### RÚBRICA PARA EVALUAR LA PARTICIPACIÓN EN CLASE

**NOMBRE DEL ESTUDIANTE:** \_\_\_\_\_  
**PARTICIPACIÓN EN CLASE.**

ELEMENTOS A EVALUAR	4 EXCELENTE	3 MUY BUENO	2 BUENO	1 REGULAR	Puntos Rúbrica
<b>Elaboración de ejercicios y programas</b>	La información recabada y el análisis, contribuyeron a obtener conclusiones profundas y significativas. <b>10</b>	La información recabada y el análisis contribuyó a obtener varias conclusiones pero sin conexión significativa con la temática. <b>8</b>	La información recabada no fue suficiente para argumentar conclusiones reveladoras. <b>6</b>	Realizó un trabajo con información insuficiente y poco útil. No incluyó conclusiones. <b>4</b>	
<b>Participación en clase</b>	Sus intervenciones mostraron gran relación con el tema expuesto. <b>10</b>	Sus intervenciones mostraron relación con el tema expuesto, pero no logró llegar a la conclusión <b>8</b>	Sus intervenciones mostraron alguna relación con el tema expuesto, pero son superficiales <b>6</b>	Sus intervenciones no muestran un análisis sobre la lectura, no se relacionan del todo con el tema. <b>4</b>	
<b>Actitud y comportamiento en clase</b>	La actitud con los compañeros y profesor fue con respeto, cordialidad y atinente y diligente a las actividades e instrucciones del profesor. <b>10</b>	El comportamiento con los compañeros y con el profesor, fue con respeto pero en algunos momentos se distraía y distraía a los demás <b>8</b>	Su actitud fue de poco interés a las actividades en clase, platicando y distraiendo a sus compañero y profesor <b>6</b>	Presentó falta de respeto a sus compañeros y/o al profesor, indisciplinado y poco atento <b>4</b>	



<b>Responsabilidad</b>	Cumplió de manera impecable con los compromisos de entrega de tareas en tiempo y forma. <b>10</b>	Cumplió a tiempo con los compromisos pero no de manera completa. <b>8</b>	Cumplió de manera impuntual, incompleta con sus compromisos. <b>6</b>	Cumplió con los compromisos sólo algunas ocasiones. <b>4</b>	
------------------------	--	--	--	---	--