Laboratorios de computación salas A y B

Profesor: García Morales Karina

Asignatura: Fundamentos de Programación

Grupo: 1121

No de Práctica(s): 5

Integrante(s): Loeza Encarnación Jafet Tonatiuh

No. de Equipo de cómputo empleado:

Semestre: 1

Fecha de entrega: 25 de septiembre de 2018

Observaciones:

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

Guía práctica de estudio 05: Pseudocódigo

Objetivo:

Elaborar pseudocódigos que representen soluciones algorítmicas empleando la sintaxis y semántica adecuadas.

¿Qué es pseudocódigo?

Un pseudocódigo es la representación escrita de un algoritmo, es decir, muestra en forma de texto los pasos a seguir para solucionar un problema. El seudocódigo posee una sintaxis propia para poder realizar la representación del algoritmo (solución de un problema).

Características del pseudocódigo:

1. Alcance del programa: Todo pseudocódigo está limitado por las etiquetas de INICIO y FIN. Dentro de estas etiquetas se deben escribir todas las instrucciones del programa.
2. Palabras reservadas con mayúsculas: Todas las palabras propias del pseudocódigo deben de ser escritas en mayúsculas.
3. Sangría o tabulación: El pseudocódigo debe tener diversas alineaciones para que el código sea más fácil de entender y depurar.

     4. Lectura / escritura: Para indicar lectura de datos se utiliza la etiqueta LEER. Para indicar escritura de datos se utiliza la etiqueta ESCRIBIR. La lectura de datos se realiza, por defecto, desde el teclado, que es la entrada estándar del sistema. La escritura de datos se realiza, por defecto, en la pantalla, que es la salida estándar del sistema.

Tipos de Datos

ENTERO -> valor entero positivo y/o negativo

REAL -> valor con punto flotante y signo

BOOLEANO -> valor de dos estados: verdadero o falso

CARACTER -> valor tipo carácter  
CADENA -> cadena de caracteres

Secuencial:

Leer: Declarar variables

Escribir: “Texto”

Operadores aritméticos:

Se tiene la posibilidad de utilizar operadores aritméticos y lógicos:

Operadores aritméticos: suma (+), resta (-), multiplicación (\*), división real (/), división entera (div), módulo (mod), exponenciación (^), asignación (:=).

Operadores lógicos: igualdad (=), y-lógica o AND (&), o-lógica u OR (|), negación o NOT (!), relaciones de orden (<, >, <=, >=) y diferente (<>).

Estructura de control secuencial

Las estructuras de control secuenciales son las sentencias o declaraciones que se realizan una a continuación de otra en el orden en el que están escritas.

PSEUDOCÓDIGO:

INICIO

X: CARACTER

ESCRIBIR: “Escoge tu helado favorito A:=Chocolate; B:=Fresa; C:=Napolitano”

LEER x

SELECCIONAR x EN

CASO 1→

ESCRIBIR “Chocolate”

CASO 2→

ESCRIBIR “Fresa”

CASO 3→

ESCRIBIR “Napolitano”

CASO 4→

ESCRIBIR “Default”

FIN SELECCIONAR

FIN

Funciones

Cuando la solución de un problema es muy compleja se suele ocupar el diseño descendente (divide y vencerás). Este diseño implica la división de un problema en varios subprocesos más sencillos que juntos forman la solución completa. A estos subprocesos se les llaman métodos o funciones.

Una función está constituida por un identificador de función (nombre), de cero a n parámetros de entrada y un valor de retorno:

INICIO

FUNC identificador (var:TipoDato,..., var:TipoDato) RET: TipoDato

[Acciones]

FIN FUNC

FIN

El identificador es el nombre con el que llama a la función. Las funciones pueden o no recibir algún(os) parámetro(s) (tipo(s) de dato(s)) como entrada; si la función recibe alguno se debe incluir entre los paréntesis. Todas las funciones pueden regresar un valor al final de su ejecución (el resultado).

Ejercicio de estructura

EJERCICIO DE ESTRUCTURA

facultad:REG

carreraAlumno:CADENA

numDeCuenta:ENTERO

nombreAlumno:CADENA

FIN REG

alu:REGfacultad

aluCiencias:REGfacultad

aluIngenieria:REGfacultad

**EJERCICIOS DE TAREA**

Ejercicio 1:

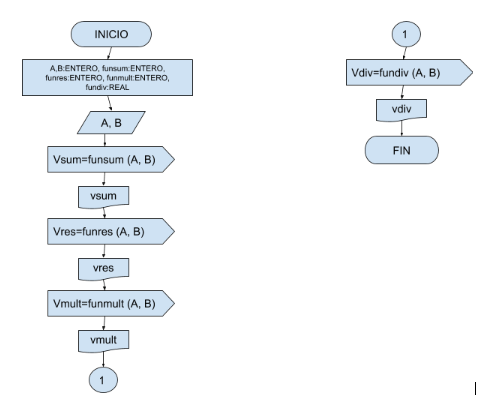
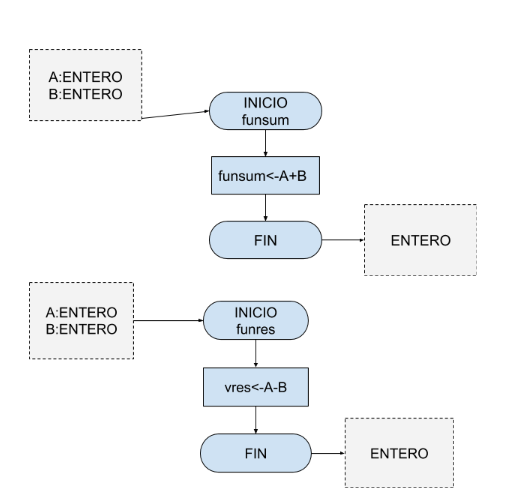
Calculadora para 2 variables (+,-,\*, /)

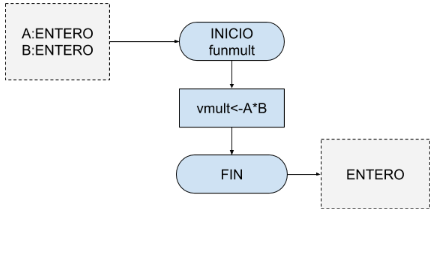
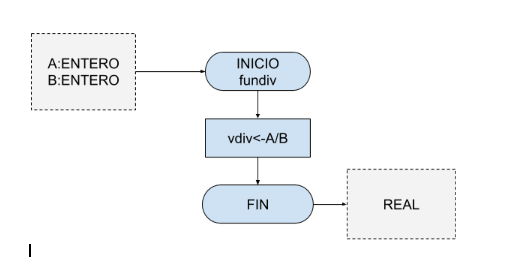
Análisis:

* Datos de Entrada: 2 números (A, B) y las funciones que se llamaran.
* Restricciones: Solo se ingresan números enteros
* Datos de Salida: Los resultados de las operaciones realizadas por las funciones llamadas.

Algoritmo:

1. Inicio
2. A, B: ENTERO, funsum: ENTERO, funres: ENTERO, funmult: ENTERO, fundiv:ENTERO
3. Ingresa valores A y B
4. Leer A y B
5. Crear las funciones para la suma, multiplicación, resta y división.
6. Llamar a la función suma y realizar operación.
7. Imprimir resultado de suma
8. Llamar a la función resta y realizar la operación
9. Imprimir el resultado de la resta
10. Llamar a la función de multiplicación y realizar operación.
11. Imprimir el resultado de la multiplicación
12. Llamar a la función división y realizar la operación
13. Imprimir el resultado de la división.
14. Fin

**DIAGRAMAS DE FLUJO Y DE FUNCIONES**



PRUEBAS DE ESCRITORIO:

|  |  |  |  |
| --- | --- | --- | --- |
| ITERACION | A | B | SALIDA |
| 1 | 4 | 2 | 6,2,8,2 |
| 2 | 16 | 8 | 24,8,128,2 |
| 3 | 10 | 5 | 15,5,50,2 |
| 4 | 21 | 7 | 28,14,147,2 |

Pseudocódigo:

Pseudocódigo

INICIO

FUNC principal

A: ENTERO Vsum:ENTERO Vres:ENTERO Vmult:ENTERO Vdiv:REAL

B: ENTERO

ESCRIBIR “Dame dos números”

LEER a

LEER b

Vsum:=(a+b)

ESCRIBIR “+” Vsum

Vres:=(a-b)

ESCRIBIR “-“ Vres

Vmult:=(a\*b)

ESCRIBIR “\*” Vmult

Vdiv:=(a/b)

Escribir “/” Vdiv

FIN FUNC

FIN

INICIO

\*\*Función de suma

FUNSUM (a:ENTERO, b:ENTERO) RET:ENTERO

Vsum:ENTERO

Vsum:=a+b

RET Vsum

FIN FUNC

FIN

INICIO

\*\*Función de RESTA

FUNSUM (a:ENTERO, b:ENTERO) RET:ENTERO

Vres:ENTERO

Vres:=a-b

RET Vres

FIN FUNC

FIN

INICIO

\*\*Función de multiplicación

FUNSUM (a:ENTERO, b:ENTERO) RET:ENTERO

Vmult:ENTERO

Vmult:=a\*b

RET Vmult

FIN FUNC

FIN

INICIO

\*\*Función de división

FUNSUM (a:ENTERO, b:ENTERO) RET:ENTERO

Vdiv:REAL

Vdiv:=a/b

RET Vdiv

FIN FUNC

FIN

Ejercicio 2

Menú de deportes (3)

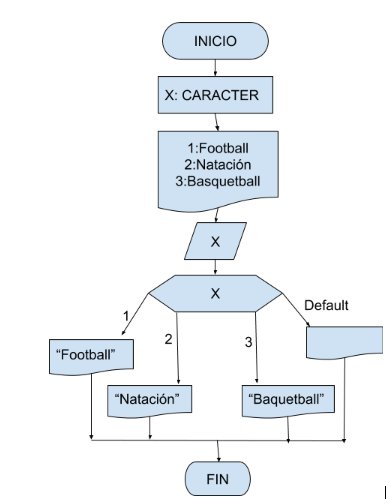
Análisis:

* Datos de Entrada: La preferencia de deportes del usuario.
* Restricciones: Ninguna.
* Datos de Salida: La elección del deporte.

Algoritmo:

1. Inicio
2. Dar el menú de las opciones con 1: Football; 2: Natación; 3: Basquetball
3. Leer y
4. Escoger cualquiera de las opciones.
5. Imprimir el deporte
6. Fin

**DIAGRAMAS DE FLUJO**



PRUEBAS DE ESCRITORIO:

|  |  |  |
| --- | --- | --- |
| ITERACIÓN | X | SALIDA |
| 1 | 2 | Natación |
| 2 | 3 | Basquetball |
| 3 | 1 | Football |

Pseudocódigo:

INICIO

x:CARACTER

ESCRIBIR:”Escoge tu helado de la siguiente lista:= 1:=”Natación”;2:=”Basquetball”;3:=”Football”

LEER:x

SELECCIONAR (x) EN

Caso 1 🡪

ESCRIBIR “Natación”

Caso 2→

ESCRIBIR “Baquetball”

Caso 3→

ESCRIBIR “Football”

Default

ESCRIBIR “Opción Inválida”

SIN SELECCIONAR

FIN

Ejercicio 3

Tablas de multiplicar del 1 al 10. El usuario proporciona el valor a calcular.

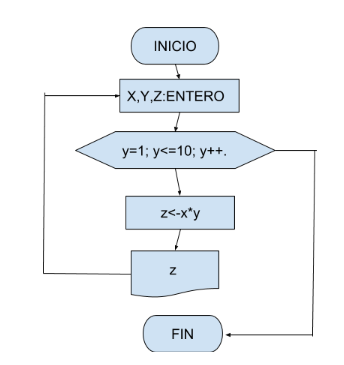
Análisis:

* Datos de Entrada: El número de la tabla de multiplicar que se desea
* Restricciones: Solo del 1 al 10
* Datos de Salida: La tabla de las multiplicaciones

Algoritmo:

1. Inicio
2. Declarar x,y,z como enteros
3. “Ingresa el valor deseado a multiplicar”
4. Leer x
5. Para y=1; y<=10; y++. Hacer, si no ir a fin.
6. Multiplicar z=x\*y
7. Imprimir z
8. Fin para
9. Fin

DIAGRAMA DE FLUJO



PRUEBAS DE ESCRITORIO:

|  |  |  |
| --- | --- | --- |
| ITERACIÓN | X | SALIDA |
| 1 | 3 | \*1=3 |
| 2 | 3 | \*2=6 |
| 3 | 3 | \*3=9 |

**Pseudocódigo:**

INICIO

x:ENTERO

y:ENTERO

z:ENTERO

ESCRIBIR “Dame el valor deseado a multiplicar”

LEER x

PARA y=1; y<=10, y++

z=x\*y

ESCRIBIR z

FIN PARA

FIN

**CONCLUSIÓN:**

Con esta práctica que llevamos a cabo pusimos en práctica todo lo que se enseñó hasta ahora. Para poder llegar después a hacer un código y entrar a fondo con lo que es programación estructural C.