

Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor:	Ing.Karina García Morales
Asignatura:	Fundamentos de programación
Grupo:	20
No. de práctica(s):	Practica 5
Integrante(s):	Francisco Javier Gómez Mendoza
No. de lista o brigada:	22
Semestre:	2023-1
Fecha de entrega:	25 de actuaré del 2022
Observaciones:	
	CALIFICACIÓN:

Pseudocódigo

Objetivo:

El alumno elaborará pseudocódigos que representen soluciones algorítmicas empleando la sintaxis y semántica adecuadas.

Introducción

Ya una vez razonado el problema y se obtienen los datos requeridos que son los de entrada y salida, y se a diseñado el algoritmo que resuelva el problema de manera eficiente, se debe pasar a la siguiente. Estaba que es codificación del algoritmo.

Para que se pueda codificar nuestro algoritmo se debe generar una réresentación de este una representación algorítmica elemental es el pseudocodigo.

El pseudocodigo es la representación escrita del algoritmo, en palabras mas entendibles son los pasos a seguir de manera de texto, posee la sintaxis para realizar la representación del algoritmo.

Sintaxis de pseudocodigo

El lenguaje pseudocódigo tiene diversas reglas semánticas y sintácticas

- 1. **Alcance del programa:** cualquier pseudocodigo va contar con dos etiquetas, INICIO y FIN, dentro de ellas van las instrucciones.
- 2. **Palabras reservadas con mayúsculas:** Las palabras propias del pseudocódigo deben de ser escritas en mayúsculas.
- 3. **Sangría o tabulación:** Debe tener diversas alineaciones para que el código sea más fácil de entender v depurar.
- 4. **Lectura / escritura:** Para indicar lectura de datos se utiliza la etiqueta LEER. Para indicar escritura de datos la etiqueta ESCRIBIR.
- 5. **Declaración de variables:** la declaración de variables la definen un identificador (nombre), seguido de dos puntos, seguido del tipo de dato.
- 6. **Operadores aritméticos:** Se tiene la posibilidad de utilizar operadores aritméticos y lógicos.
- 7. **Notación de camello**. Para nombrar variables y nombres de funciones se debe hacer uso de la notación de camello.

Estructuras de control de flujo

Permiten la ejecución condicional y la representación de un conjunto de instrucciones, existen 3 tipos, secuencial, condicional y repetitivas o iterativas.

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.

Ejemplo

```
INICIO

x : REAL

x := 5.8

x := x * 2
```

Estructuras de control condicionales.

Permiten evaluar una expresión lógica, esta puede ser falsa o verdadera, dependiendo del resultado, se realiza uno o otro flujo de instrucciones. Estás estructuras son mutuamente excluyentes o se realiza una acción o otra.

La estructura de control de flujo más simple es la estructura condicional SI, su sintaxis es la siguiente:

```
SI condición ENTONCES
[Acciones]
FIN SI
```

Estructuras de control iterativas o repetitivas.

Las estructuras de control de flujo iterativas o repetitivas, permiten realizar una serie de instrucciones mientras se cumpla la expresión lógica. Existen 2 tipos MIENTRAS y HACER- MIENTRAS.

MIENTRAS. Primero valida la condición y si ésta es verdadera y asi puedea realizar las instrucciones de la estructura y regresa a validar la condición, hasta que es falsa rompe el ciclo. El fin de la estructura lo determina la etiqueta FIN MIENTRAS.

```
Ejemplo

INICIO

valorInicial,valorFinal:ENTERO
valorFinicial:=0
valorFinal:=3

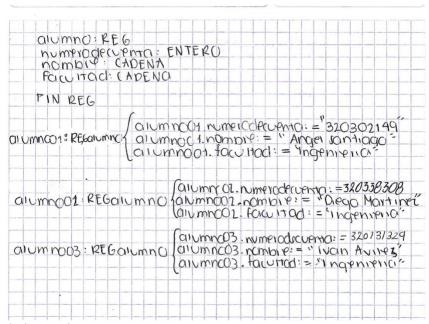
MIENTRAS valorInicial < valorFinal
ESCRIBIR valorInicial
valorInicial := valorInicial + 1

FIN MIENTRAS
```

HACER-MIENTRAS. primero realiza las instrucciones descritas en la estructura y después valida la expresión lógica, si la condición se cumple vuelve a realizar las instrucciones de la estructura, de lo contrario rompe el ciclo y sigue el flujo del pseudocódigo.

TRABJO EN CLASE

1. En clase resolvimos el problema de recaudar datos con registro, en mi caso pedía información escolar sobre alumnos, estos datos que junte son datos de mis compañeros de clase.



2. Segundo trabajo en clase.

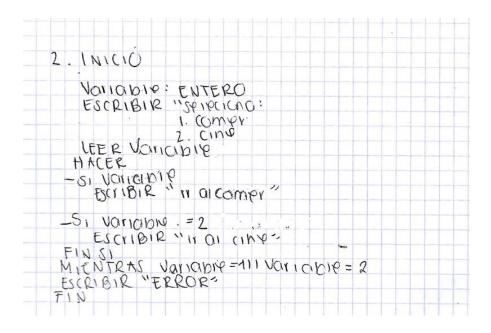


TAREA

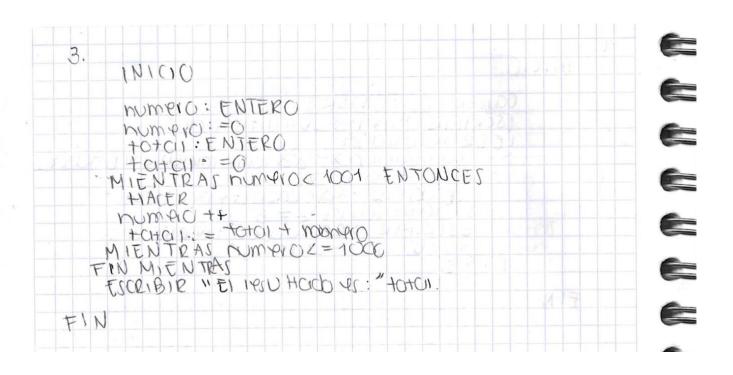
1.- Algoritmo que valide la calificación que ingrese el usuario, esta debe estar entre 5 y 10. Indicar si ha aprobado con calificación mayor a 6;un letrero "Aprobado, felicidades", en caso de no aprobar "Reprobado, tienes una nueva oportunidad"

1.	
	INICIO
	Canticion: ENTERO Escribir "imprere ju continación" LE ER Continación SI canticación > 4 && canticación < 11 Entorces - SI canticación > 6 Escribir "Aprobach, tenícidadus" - SI canticación & = 6 Escribir "Repropodo tienes una
	· FINSI COPULTUNICION
	FIN

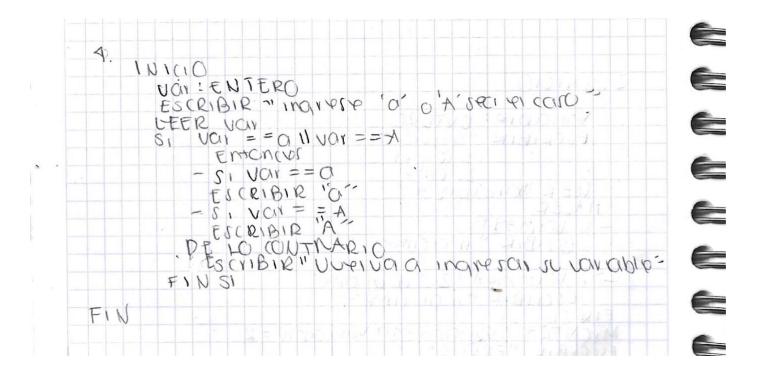
2.- Del ejercicio de clase (el menú) modifica con el ciclo hacer mientras, en donde valide, si el usuario no elige las opciones 1ó 2 para caracteres deben ser 'a', 'b' o 'c') imprimir un letrero de "error"(es el default) y volver a mostrarle el menú..



3.-Algoritmo que muestre la numeración del 1 al 1000, emplea la estructura MIENTRAS, HA-CER MIENTRAS Y PARA



4.- Solicitar al usuario que ingrese la variable, si ingresa una variable diferente a la letra 'a' o 'A', volver a solicitarla, en caso de contrario, imprimir la letra 'a' o 'A' que ingresó el usuario.



Conclusiones

- 1. Esta ovación la pre¿activa se torno más difícil, tal vez por que la clase fue en línea y quedaron lagunas de cosas, pero tuve que pedir ayuda de mis compañeros para resolver dudas y un poco de ayuda.
- **2.** El pseudocodigo es un poco complicado para mi, pues la notación de como acomodar los datos se me hace aún difícil
- 3. El apoyo de los digamma de flujo es in fluyente, me gustaría usarlos en los siguientes ejercicios pero esta vez me queda como lección esto.
- **4.** También empezamos a ver como funciona el peso del código en cómo hacer la notación de sintaxisY así para la siguiente etapa que es la programación sea mucho más fácil y no tener problemas en el proceso.
- **5.** En lo personal es la practica que menos me a gustado pero no estuvo tan mal, creo que de todo se aprende y fue el caso. Pero me gusta que todo se mas didáctico.

Bibliografía

Laboratorio Salas A y B. (2022, 21 febrero). Manual de prácticas del Laboratorio de Fundamentos de programación. Recuperado 28 de agosto de 2022, de http://lcp02.fi-b.unam.mx/

Liga de github

https://github.com/JavierMend/practica1_fdp/blob/e057ebb63f415fa39e11545dfa3ee774 25a03e12/Practica 1.pdf