



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: Karina Morales Garcia

Asignatura: Fundamentos de programacion

Grupo: 20

No. de práctica(s): 4

Integrante(s): Avila Pineda Samuel David

No. de lista o brigada: 06

Semestre: 2023-1

Fecha de entrega: 25 de octubre del 2022

Observaciones:

CALIFICACIÓN: _____

Objetivo

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

Desarrollo

Una clase antes de la práctica de laboratorio vimos todo lo “teórico” esto con la finalidad de que el día de la práctica solo fuera puros ejercicios, sin embargo, si se vieron diferentes cosas importantes, que a continuación mostrare, vimos la sintaxis y sus respectivos ejemplos de cómo se vería reflejado en el pseudocódigo. Antes de iniciar bien con la redacción de esta práctica es importante definir ¿Qué es el Pseudocódigo?

¿Qué es el Pseudocódigo?

Se podría decir que es la representación escrita de un algoritmo o en otras palabras son los pasos a seguir para solucionar un problema y dichos pasos se expresan de manera escrita.

Sintaxis del Pseudocódigo

El lenguaje del pseudocódigo contiene diversas reglas semánticas y sintácticas, por ejemplo:

1. Alcance del programa: Esta limitado, lo que quiere decir que tiene un INICIO y un FIN.
2. Palabras reservadas con mayúsculas: Las palabras propias del Pseudocódigo deben de estar escritas en mayúsculas.
3. Sangría o tabulación: Debe de tener diversas alineaciones, esto con el fin de hacerlo más entendible.
4. Lectura / escritura: Para indicar lectura de datos se utiliza la etiqueta LEER y para indicar escritura de dichos datos se utiliza ESCRIBIR.

```
ESCRIBIR "Ingresar la altura del polígono"  
LEER altura
```

5. Declaración de variables: Es un identificador (nombre) seguido de dos puntos, seguido del tipo de dato:

```
<nombreVariable>:<tipoDeDato>
```

Existen 5 tipos de datos:

Entero – es un valor positivo o negativo.

Real – valor con punto flotante y signo.

Booleano – valor de dos estados: verdadero falso.

Carácter – valor tipo carácter.

Cadena – cadena de caracteres.

```
contador: ENTERO  
producto: REAL  
continuar: BOOLEANO
```

Se puede declarar más de una variable de un mismo tipo de dato utilizando arreglos, su sintaxis sería la siguiente.

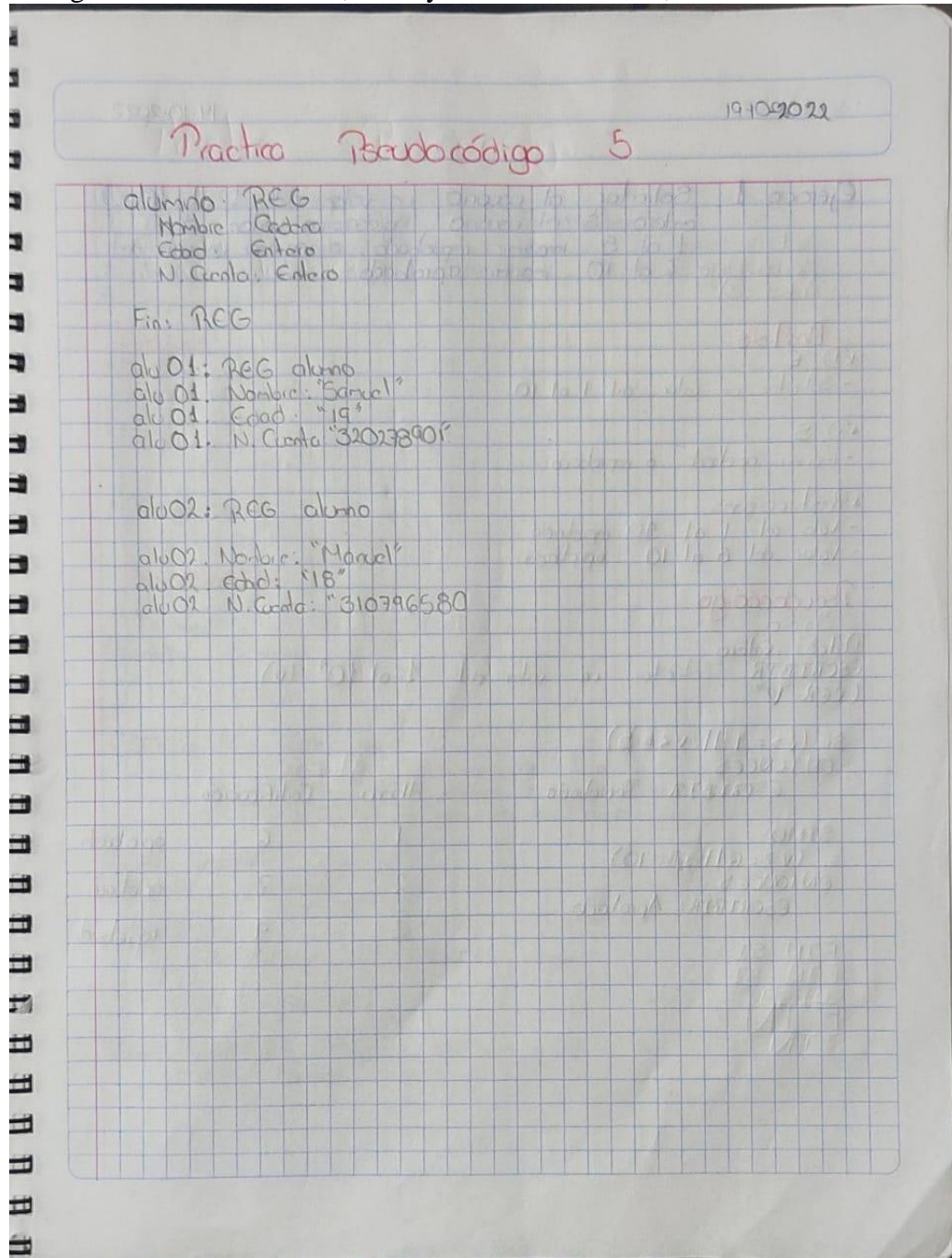
```
<nombreVariable>[cantidad]:<tipoDeDato>
```

Existe un tipo de dato compuesto, el cual puede contener uno o más tipos de datos simples diferente, se conoce como registro o estructura. La sintaxis es:

```
<nombreRegistro>:REG  
  <nombreVariable_1>:<tipoDeDato>  
  ...  
  <nombreVariable_N>:<tipoDeDato>  
FIN REG
```

Para crear una variable tipo registro se debe indicar el nombre del registro y nombre de la variable. Se utiliza el punto para acceder a los datos del registro.

A continuación, mostrare una imagen del ejercicio que se elaboró a la hora de la práctica, teníamos que crear una variable de tipo registro con el nombre "alumno" con los siguientes datos Nombre, Edad y Número de cuenta, se hizo en total 3.



Se puede crear variables constantes con la palabra reservada CONST, esto nos sirve para indicar que un identificador no cambia su valor durante todo el pseudocódigo.

NUM_MAX := 1000: REAL, CONST

6. Operadores aritméticos: se pueden utilizar operadores aritméticos y lógicos.

Aritmeticos: suma (+), resta (-), multiplicación (*), división (/), modulo (mod), exponenciación (^), asignación (:=).

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

7. Notación de camello: Se utiliza para nombrar variables y nombres de funciones.

Su nombre proviene de que los nombres de cada palabra empiezan con mayúscula y el resto con minúscula. Hay dos tipos, lower camel, la prmer letra de la variable inicia con minúscula y la upper camel todas las palabras inician con mayúscula.

```
realAreaDelTriangulo: REAL    → lower camel case
EnteroRadioCirculo: REAL     → upper camel case

calcularArea()
obtenerPerimetro()
```

Estructuras de control de flujo

Estas nos ayudan a la ejecución condicional y la repetición de un conjunto de instrucciones, hay 3 tipos, las cuales son: secuencial, condicional repetitivas o iterativas

Estructura de control secuencial.

Su nombre lo dice, son aquellas que se realizan una a continuación de otra.

```
INICIO
    x : REAL
    x := 5.8
    x := x * 2
FIN
```

Estructura de control condicional.

Estas nos permiten evaluar una expresión lógica y dependiendo del resultado, se realiza uno u otro flujo de instrucciones. La estructura más simple es la condicional.

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

Primero se evalúa y si se cumple se realizan las siguientes instrucciones.

```
INICIO
    a,b: ENTERO
    a := 3
    b := 2
    SI a > b ENTONCES
        ESCRIBIR "a es mayor"
    FIN SI
FIN
```

Estructura de control repetitivas o iterativas.

También son llamadas cíclicas y nos permite realizar una serie de instrucciones siempre y cuando se cumpla la expresión lógica. Hay dos tipos: MIENTRAS y HACER – MIENTRAS.

En la primera, lo primero que se hace es validar la condición y si es verdadera siguen con las siguientes instrucciones.

```
MIENTRAS condición ENTONCES
    [Acciones]
FIN MIENTRAS
```

En la segunda es lo contrario, primero se realiza las instrucciones y luego se valida.

```
HACER
    [Acciones]
MIENTRAS condición
```

Ejercicios en clase:

19-10-2022

Ejercicio 1. Solicitar al usuario un valor del 1 al 10 entero, si el usuario ingresa un valor del 1 al 5 imprimir reprobado, si ingresa un valor del 6 al 10 imprimir aprobado.

Análisis: 1 al 5 imprimir reprobado, si ingresa un valor del 6 al 10 imprimir aprobado

Pseudocódigo: 6 al 10 imprimir aprobado

Pruebas (3)

Análisis

- *1. E
- Solicitar un valor del 1 al 10
- *1. S
- imprimir aprobado o reprobado
- *Restricciones
- Valor del 1 al 5 reprobado
- Valor del 6 al 10 aprobado

Pseudocódigo

Inicio

Datos: Entero

ESCRIBIR: "Solicitar un valor del 1 al 10" (v)

LEER "V"

Si $(v >= 1 \vee v <= 5)$

ENTONCES

ESCRIBIR: Reprobado

Alumno	Calificación	Aprobado o reprobado
1	6	aprobado
2	7	aprobado
3	4	reprobado

FINO

Si $(v >= 6 \vee v <= 10)$

ENTONCES

ESCRIBIR: Aprobado

FIN SI

FIN

FIN

Ejercicio 2 Genera un menú

1. Imprime los números del 1 al 5 (Mientras)
2. Pregunta al usuario si va al cine o a comer
3. Imprime la suma de los números del 10 al 1
- 4.

Análisis

- * D.E
Imprimir los números del 1 al 5 (Mientras), Pregunta al usuario si va al cine o a comer, Imprimir suma de los números del 10 al 1
- * D.S
Generar un menú con los datos que nos proporcionaron

* Restricciones

No hay

Pseudocódigo

Inicio

Variable: Entero
Carácter
Entero

ESCRIBIR "Seleccionar"

1. Imprime los números del 1 al 5
2. Pregunta al usuario si va al cine o a comer
3. Imprime la suma de los números del 10 al 1

Leer Variable (Entero)

Seleccionar Variable

Caso 1:

num = 1

MIENTRAS (num < 5) ENTONCES

Imprimir num

FIN MIENTRAS

Caso 2:

ESCRIBIR "¿vas al cine o a comer?"

Leer Cadena

Caso 3:

ESCRIBIR "Imprime la suma de los números de 10 al 1"

variable: Enteros contador

a ← 10 contador ← 1

b ← 11

Si (a < b) ENTONCES

contador + a

19-10-2022

Imprimir
SI NO
Escribir "la suma de los números del 10 al 12 es"
FIN MIENTRAS
FIN

Iteración	Opción elegida	Resultado
1	Caso 1 y resultado	✓
2	Caso 2 y resultado	✓
3	Caso 3 y resultado	✓

TAREA:

24-10-2022

TAREA

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

Analisis

* D.E

Ingresar calificación

* D.S

"Aprobado, felicidades", y "Reprobado, tienes una nueva oportunidad"

* Restricciones

Debe ser entre 5 y 10, calificación aprobatoria es mayor a 6

Algoritmo

1. Inicio
2. El usuario ingresa una calificación (num. entero)
3. Leer calificación (var. calificación)
4. Si $\text{var. Calificación} \geq 5$ y $\text{var. Calificación} \leq 10$ pasar al siguiente paso, sino regresar al paso 2
5. Si $\text{var. Calificación} \leq 6$ se imprime "Reprobado, tienes una nueva oportunidad" y pasar al paso 7
6. Si $\text{var. Calificación} > 6$ y $\text{var. Calificación} \leq 10$ se imprime "Aprobado, felicidades" y pasar al siguiente paso
7. Fin

Prueba de escritorio

Iteración	Datos de entrada	Datos de salida	Resultado
1	5	"Reprobado, tienes una nueva oportunidad"	✓
2	9	"Aprobado, felicidades"	✓
3	7	"Aprobado, felicidades"	✓

24-10-2022

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

Análisis

* 1. E

1-2

* 1. S

Cada opción contiene diferente salida

* Testeo

Si el usuario no escoge opción 1 o 2 imprimir "error"

Inicio

Número Entero

Variable Entero

ESCRIBIR * Selecciona que opción desplega en estas cosas

1. Imprimir los números del 1 al 5
2. Preguntar al usuario si va al cine o a comer
3. Imprimir la suma de los números del 1 al 10

Si (var. número = 1) el var. número (≤ 3) pasar a la opción después
MIENTRAS (var. número = 0) el var. número (≤ 3) volver al menú
var. número = 1 = caso 1

Selecciona caso 1

MIENTRAS (var. número ≤ 5) ENTONCES

Imprimir var. número

Repetir

Preguntar al MIENTRAS - Si var. número ≤ 5 SI NO

FIN MIENTRAS

LEER Var. número

Si var. número = 2

Selecciona caso 2

ESCRIBIR * Selecciona la opción a realizar

LEER var. número

24-10-2012

Imprimir si varianza = 1: va al cab
 Imprimir si varianza = 2: va a com
 FIN del Si

FIN

Pruebas de escritorio

Iteración	D.C	D.S	Resultado
1	2	"Proceso del 2"	✓
2	4	"Proceso al mal"	X
3	1	"Proceso al mal"	X

3. Algoritmo que muestre la numeración del 1 al 1000, aplica la estructura MIENTRAS, HACER MIENTRAS y PARA

Análisis

- * D.E
- * Obtener una numeración
- * D.S
- * Numeración del 1 al 1000
- * Restricciones
- * Solo del 1-1000

Algoritmo MIENTRAS

Inicio MIENTRAS
Definir n, contador como un entero
contador = 1
MIENTRAS Contador \leq 1000 Hacer
Escribir contador
Contador = contador + 1
FIN MIENTRAS
FIN

Pruebas de escritorio

Iteración	D.E	D.S	R
1	1001	Fin	X
2	27	contador	/
3	6	contador	/

Algoritmo HACER MIENTRAS

Inicio HACER MIENTRAS
Definir n, como Entero
i = 1
Escribir i
Leer i
Repetir
i = i + 1
Escribir hasta que i \leq 100
FIN Repita
FIN

Pruebas de escritorio

Iteración	D.S	D.E	R
1	Conteo	4	/
2	Conteo	100	/
3	Fin	1001	X

Algoritmo PARA

Inicio PARA
ni
Definir i como entero
Escribir n
Leer n
i = 1
PARA (i = i + 1) Hasta (i \leq 1000)
FIN PARA
FIN

Pruebas de escritorio

Iteración	D.E	D.S	R
1	1009	Fin	X
2	1110	Fin	X
3	1137	Fin	X

25-10-2022

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 ingreso el usuario

Análisis

* D.E

Ingresar variable, Imprimir variable

* D.S

Imprimir variable de la letra "a" o "A"

* Restricciones

Para imprimir, debe ser igual a la letra "a" o "A" en caso de que sea otra letra, volver a solicitar variable

Pseudocódigo

INICIO

ESCRIBIR "Ingresar una letra" varCaracter

LEER varLetra

SI varCaracter <- a || varLetra <- A)

ENTONCES

ESCRIBIR "a" || "A"

SINO

ESCRIBIR "Ingresar otra letra"

FIN SI

FIN

Pruebas de escritorio

Iteración	D.E	D.S	Resultado
1	A	A	✓
2	a	a	✓
3	d	"Ingresar otra letra"	✗

REFERENCIAS:

Metodología de la programación. Osvaldo Cairó, tercera edición, México D.F., Alfaomega 2005.

Metodología de la programación a través de pseudocódigo. Miguel Ángel Rodríguez Almeida, primera edición, McGraw Hill