|  |  |  |
| --- | --- | --- |
|  | **Carátula para entrega de prácticas** | |
| Facultad de Ingeniería | | Laboratorio de docencia |

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Claudia Rodriguez Espino |
| *Asignatura:* | Fundamentos de Programación |
| *Grupo:* | 4 |
| *No de Práctica(s):* | 5 |
| *Integrante(s):* | Sanchez Escamilla Hector |
|  |  |
| *No. de Equipo de cómputo empleado* | 35 |
| *Semestre:* | 2 |
| *Fecha de entrega:* | 15/03/2018 |
| *Obervaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

Práctica 05: Pseudocódigo

**Objetivo:**

Elaborar pseudocódigos que representen soluciones algorítmicas empleando la sintaxis y

semántica adecuada.

**Actividades:**

Elaborar un pseudocódigo que represente la solución algorítmica de un problema

en el cual requiera el uso de la estructura de control de flujo condicional.

A través de un pseudocódigo, representar la solución algorítmica de un problema

en el cual requiera el uso de la estructura de control iterativa.

**Introducción**

Una vez que un problema dado ha sido analizado (se obtiene el conjunto de datos de

entrada y el conjunto de datos de salida esperado) y se ha diseñado un algoritmo que lo

resuelva de manera eficiente (procesamiento de datos), se debe proceder a la etapa de

codificación del algoritmo.

Para que la solución de un problema (algoritmo) pueda ser codificada, se debe generar

una representación del mismo. Una representación algorítmica elemental es el

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 pseudocódigo posee una

sintaxis propia para poder realizar la representación del algoritmo (solución de un

problema).

**Sintaxis de pseudocódigo**

El lenguaje pseudocódigo tiene diversas reglas semánticas y sintácticas.

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.

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:

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 (<>).

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**

Las estructuras de control de flujo permiten la ejecución condicional y la repetición de un

conjunto de instrucciones.

Existen 3 estructuras de control: 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.

**Estructuras de control condicionales (o selectivas)**

Las estructuras de control condicionales permiten evaluar una expresión lógica (condición

que puede ser verdadera o falsa) y, dependiendo del resultado, se realiza uno u otro flujo

de instrucciones. Estas estructuras son mutuamente excluyentes (o se ejecuta una acción o

se ejecuta la otra).

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

la siguiente:

SI condición ENTONCES

[Acción]

FIN SI

La estructura condicional completa es SI-DE LO CONTRARIO:

SI cond\_booleana ENTONCES

[Acciones SI]

FIN SI

DE LO CONTRARIO

[Acciones DE LO CONTRARIO]

FIN DE LO CONTRARIO

La estructura condicional SELECCIONAR-CASO valida el valor de la variable que está

entre paréntesis, y comprueba si es igual al valor que está definido en cada caso. Si la

variable no tiene el valor de ningún caso se va a la instrucción por defecto (DEFECTO).

SELECCIONAR (variable) EN

CASO valor1 -> [Acción]

CASO valor2 -> [Acción]

CASO valor3 -> [Acción]

DEFECTO -> [Acción]

FIN SELECCIONAR

**Estructuras de control iterativas o repetitivas**

Las estructuras de control de flujo **iterativas o repetitivas** (también llamadas cíclicas)

permiten ejecutar una serie de instrucciones mientras se cumpla la expresión lógica.

Existen dos tipos de expresiones cíclicas MIENTRAS y HACER- MIENTRAS.

**Actividad:**

1. Calcular el área de un círculo

INICIO

ESCRIBIR “Ingrese el radio del círculo”

LEER radio

a : REAL

radio : REAL

P :REAL

P: =3.1416

a :P\*(r\*r)

ESCRIBIR “El resultado es: a”

FIN

      2. Tablas de multiplicar del 1 al 10

INICIO

ESCRIBIR “Dame un número: “

LEER x

x : ENTERO

cont : ENTERO

cont :=1

m : ENTERO

PARA cont≤10 HACER

m :x\*cont

cont :=cont+1

ESCRIBIR “El resultado es: m“

FIN PARA

FIN

      3.Ecuaciones

INICIO

x : ENTERO

y : ENTERO

ESCRIBIR “Dame un numero diferente de 2”

LEER x

SI x> 2 ENTONCES

y: = 3x^2+3x-25

FIN SI

DE LO CONTRARIO

y: =2x^2-3x+3

FIN DE LO CONTRARIO

ESCRIBIR “El resultado es : y”

FIN

4. Menú de altas y bajas

INICIO

a: ENTERO

a:=1

SELECCIONAR (a) EN

CASO 1 ->

ESCRIBIR “Se encuentra en altas”

CASO 2 ->

ESCRIBIR “Se encuentra en bajas”

CASO 3 ->

ESCRIBIR “Se encuentra en cambios”

DEFECTO ->

ESCRIBIR “No existe opción”

FIN DE SELECCIONAR

FIN

5. Imprimir números del 1 al 100 con while

INICIO

x: ENTERO

x:=1

MIENTRAS x≤100

ESCRIBIR x

x: =x+1

FIN PARA

FIN

5.1. Imprimir números del 1 al 100 con do while

INICIO

x: ENTERO

x:=1

HACER

ESCRIBIR X

x:=x+1

MIENTRAS x≤100

FIN

Conclusiones:

Es un lenguaje muy util en el desarrollo de algoritmos. En pseudocodigo se describen los algoritmos utilizando una mescla de lenguaje común, con instrucciones de programación, palabras claves etc. Creo que tiene como objetivo que el programador solo se centre en la solucón lógica del algoritmo y no en la implementación en un lenguaje de programación concreto (con las posibles complicaciones en las reglas sintácticas), o en otras palabras, solo ayudan a pensar un programa antes de escribirlo en un lenguaje de programación formal.