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

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Ing. Claudia Rodríguez Espino |
| *Asignatura:* | Fundamentos de Programación |
| *Grupo:* | 1102 |
| *No de Práctica(s):* | 05 |
| *Integrante(s):* | Alejandro Meneses Mercado |
|  |  |
|  |  |
| *Semestre:* | 2018-1 |
| *Fecha de entrega:* | 15 de septiembre del 2017 |
| *Observaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

Práctica 05: Pseudocódigo

Objetivo:

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

Desarrollo:

Primero vimos que para que la solución de un problema pueda ser codificada, se debe generar una representación del mismo. Una representación algorítmica elemental es el pseudocódigo.

Vimos que el pseudocódigo es la representación escrita de un algoritmo, que muestra en forma de texto los pasos a seguir para solucionar un problema.

También vimos que el pseudocódigo posee una sintaxis propia para poder realizar la representación del algoritmo, la cual es la sig:

Lo primero es que todo pseudocódigo está limitado por las etiquetas de INICIO y FIN. Dentro de estas etiquetas se deben escribir todas las instrucciones del programa.

Otra cosa es que todas las palabras propias del pseudocódigo deben de ser escritas en mayúsculas.

Algo importante es que el pseudocódigo debe tener diversas alineaciones para que el código sea más fácil de entender y depurar.

Vimos que para indicar lectura de datos se utiliza la etiqueta LEER y para indicar escritura de datos se utiliza la etiqueta ESCRIBIR.

Un ejemplo de lo anterior es:

ESCRIBIR “Ingresar el radio del circúlo”

Leer “radio”

Otra cosa muy importante es la declaracion de variables las cuales se definen mediante un identificador,dos puntos y tipo de dato.

Un ejemplo es:

a: ENTERO

Después vimos un poco de arreglos que nos permiten declarar más de una variable de un mismo tipo de dato, indicando la cntidad de variables que se requieren.

Un ejemplo es el sig:

contador[5]: ENTERO

Otra cosa que vimos fue el Registro el cual puede contener uno o más tipos de datos simples diferentes.

Lo siguiente que vimos fueron los operadores aritmeticos los cuales son:

suma (+), resta (-), multiplicación (\*), división real (/), división

entera (div), módulo (mod), exponenciación (^), asignación (:=).

A continuación vimos los operadores lógicos los cuales son:

igualdad (=), y-lógica o AND (&), o-lógica u OR (|), negación o

NOT (!), relaciones de orden (<, >, <=, >=) y diferente (<>).

También vimos la notación de camello en la cual los nombres de cada palabra empiezan con mayúscula y el resto se escribe con minúsculas.

Vimos que existen 2 tipos de notación de camello :

La primera es lower camel case en la cual la primera letra de la variable inicia con minúscula y upper camel case en la cual todas las palabras inician con mayúscula.

Retomando conceptos de la práctica anterior vimos las estructuras de control de flujo, que son 3:

La primera es la estructura de control secuencial

Que nos dice que son las sentencias o declaraciones que se realizan

una a continuación de otra en el orden en el que están escritas.

Después vimos la estructuras de control condicionales que nos permiten evaluar una expresión lógica que es una condición que puede ser verdadera o falsa y, dependiendo del resultado, se realiza uno u otro flujo de instrucciones.

Vimos que la estructura de control de flujo más simple es la estructura condicional SI y vimos su sintaxis.

En esta estructura condicional se evalúa la expresión lógica y si se cumple se ejecutan las instrucciones del bloque [Acción]. Si no se cumple la condición, se continúa con el flujo normal del programa.

También vimos la estructura condicional SI-DE LO CONTRARIO, donde se evalúa la expresión lógica y si se cumple se ejecutan las instrucciones del bloque SI [Acciones SI]. Si no se cumple la condición se ejecutan las instrucciones del bloque DE LO CONTRARIO [Acciones DE LO CONTRARIO].

Otra estructura condicinal que vimos fue la de SELECCIONAR-CASO en la cual 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.

Después vimos las estructuras de control de flujo iterativas o repetitivas que nos permiten ejecutar una serie de instrucciones mientras se cumpla la expresión lógica.

Vimos que hay 2 estructuras y la primera es la estructura MIENTRAS donde primero valida la condición y si ésta es verdadera procede a ejecutar el bloque de instrucciones de la estructura, de lo contrario rompe el ciclo y continúa el flujo normal del pseudocódigo.

La segunda estructura es la estructura HACER-MIENTRAS donde primero se ejecutan las instrucciones descritas en la estructura y al final se valida la expresión lógica. Si la condición se cumple vuelve a ejecutar las instrucciones de la estructura, de lo contrario rompe el ciclo y sigue el flujo del pseudocódigo.

Por ultimo vimos que cuando la solución de un problema es muy compleja se suele ocupar el diseño descendente. Este diseño implica la división de un problema en varios subprocesos más sencillos que juntos forman la solución completa y a estos subprocesos se les llaman módulos o funciones.

Vimos que las funciones estan constituidas por un identificador de función, de cero a n parámetros de entrada y un valor de retorno.

Como actividad tuvimos que hacer 4 pseudocódigos de programas vistos en la práctica anterior:

1. Programa: Formación de triangulos

INICIO

a, b, c:ENTERO

ESCRIBIR “Digite el valor de a, b y c”

LEER a, b, c

SI (a=b) y (b=c) y (c=a) ENTONCES

ESCRIBIR “El triángulo es equilátero”

FIN SI

DE LO CONTRARIO

SI (a=b) y (b ≠c) y (c ≠a) ENTONCES

ESCRIBIR “El triángulo es isósceles”

FIN SI

DE LO CONTRARIO

ESCRIBIR “El triángulo es escaleno”

FIN DE LO CONTRARIO

FIN DE LO CONTRARIO

FIN

1. Programa: Solución de ecuaciones

INICIO

y,x:Entero

ESCRIBIR “Digite el valor de y”

LEER y

SI (y>2) ENTONCES

x=-3y+0

FIN SI

DE LO CONTRARIO

SI (y<2) ENTONCES

x=-4y-25

FIN SI

DE LO CONTRARIO

ESCRIBIR “No hay solución”

FIN DE LO CONTRARIO

FIN DE LO CONTRARIO

FIN

1. Programa: Suma de dos numeros

INICIO

a,b,c:ENTERO

ESCRIBIR “Digite el valor de a,b,c”

LEER a,b,c

a+b

SI a+b=c ENTONCES

ESCRIBIR “La suma del primer numero mas el segundo es igual al tercer numero”

FIN SI

DE LO CONTRARIO

ESCRIBIR “La suma del primer numero mas el segundo es diferente del tercer numero”

FIN DE LO CONTRARIO

FIN

1. Programa: Resolver la fórmula general

INICIO

a,b,c,d,x1,x2:REAL

ESCRIBIR “Digita el valor de a”

LEER a

SI (a=0) ENTONCES

ESCRIBIR “a debe ser distinto de cero”

FIN SI

DE LO CONTRARIO

d=

FIN DE LO CONTRARIO

LEER d

SI (d<0) ENTONCES

x1= +i

x2= -i

FIN SI

DE LO CONTRARIO

x1=

x2=

FIN DE LO CONTRARIO

FIN

Conclusiones:

En esta práctica retomamos los conceptos de algoritmo y diagrama de flujo para poder llegar a la definicion del pseudocódigo, el cual es la representación elemental del algoritmo. Gracias a esta práctica entendimos mejor la sintaxis para poder empezar a programar.