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

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Rodriguez Espino Claudia |
| *Asignatura:* | Fundamentos de Programacion |
| *Grupo:* | 1102 |
| *No de Práctica(s):* | Practica numero 5 |
| *Integrante(s):* | Gutierrez Orozpe Luis Fernando |
|  |  |
|  |  |
| *Semestre:* | 1er Semestre. |
| *Fecha de entrega:* | 17 de Septiembre de 2017 |
| *Obervaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

OBJETIVO

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

ACTIVIDADES

Para poder realizar los pseudocodigos que el objetivo nos pide, es necesario conocer la sintaxis de un pseudocodigo, por lo tanto, se mostrara a continuación una breve explicación y ejemplificación de un pseudocodigo.

**PSEUDOCODIGO**

Una vez que un problema dado ha sido analizado (se obtiene el conjunto de datos de entra y el conjunto de datos de salida esperad) 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 pseudocodigo.

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

**SINTAXIS DE PSEUDOCODIGO.**

El lenguaje pseudocodigo tienes diversas reglas semánticas y sintácticas. A continuación, se describen las más importantes:

1. **Alcance del programa**: Todo pseudocodigo 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 pseudocodigo deben ser escritas en mayúsculas.
3. **Sangría o tabulación**: El pseudocodigo 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 indicador (nombre), seguido de dos puntos, seguido de tipo de dato, es decir.

<nombreVariable>:<tipoDeDato>

Los tipos de datos que se pueden utilizar son:

Entero 🡪 valor entero positivo y/o negativo

Real 🡪 valor con punto flotante y signo

Booleano 🡪 valor de dos estados: verdadero o falso

Carácter 🡪 valor tipo carácter

Cadena 🡪 cadena de caracteres

1. **Operadores aritméticos**: Se tiene la posibilidad de utilizar operadores aritméticos y lógicos:

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

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

1. **Notación de camello**. Para nombrar variables y nombres de funciones se debe hacer uso de la notación de camello.

En la notación de camello, los nombres de cada palabra empiezan con mayúscula y el resto se escribe con minúsculas. Existen dos tipos de notaciones de camello: lower camel case que 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.

No se usan puntos ni guiones para separar las palabras (a excepción de las constantes que utilizan guiones bajos). Además, para saber el tipo de variable se recomienda utilizar un prefijo.

Así mismo, al igual que en la práctica de Diagramas de Flujo, en los pseudocodigos también se utilizan las estructuras de control de flujo, dentro de estas estructuras podemos encontrar tres tipos los cuales son:

Estructuras de control condicionales (o selectivas)

Estructuras de control iterativas o repetitivas

Estructuras de control secuencial.

La única diferencia entre esta práctica y la anterior, es la manera en la que la información aparece, ya que el diagrama de flujo es la representación gráfica de un algoritmo, mientras que los pseudocódigos son la representación en forma de texto de un algoritmo.

**PSEUDOCODIGOS.**

Programa 1: Chicharronera

INICIO

Variable a,b,c: Entero

Variable d,z,r,n, x1,x2: Real

ESCRIBIR: “escribe el valor de la variable a, debe de ser diferente de 0”

LEER: a

SI a=0 ENTONCES

ESCRIBIR: “No es posible porque se crea una indeterminación”

FIN SI

DE LO CONTRARIO

LEER: b, c

d = b2 – 4ac

SI d < 0 ENTONCES

z =

r = **- b / 2a**, **n=z/2ª**

x1 = r + ni

x2 = r – ni

ESCRIBIR: “Tus raíces imaginarias son”, x1, x2

FIN SI

DE LO CONTRARIO

x1 =

x2 =

ESCRIBIR: “Tus raíces son”, x1, x2

FIN DE LO CONTRARIO

FIN DE LO CONTRARIO

FIN

El programa anterior nos pedía realizar el pseudocódigo para un programa que nos permita realizar la ecuación general o (chicharronera) a modo de poder obtener tanto raíces reales como imaginarias.

**PROGRAMA 2: ECUACIONES**

INICIO

Variable y, x: Entero.

ESCRIBIR “Introduce el valor de la variable y”

LEER: y

SI y < 2 ENTONCES

x = y2 + 4y – 25

ESCRIBIR: “El valor de x es”, x

FIN SI

DE LO CONTRARIO

SI y > 2 ENTONCES

x = 4y2 – 3y + 0

ESCRIBIR “El valor de x es”, x

FIN SI

DE LO CONTRARIO

ESCRIBIR “No hay solución para y=2”

FIN DE LO CONTRARIO

FIN DE LO CONTRARIO

FIN

En el programa anterior nos pide que podamos resolver dos ecuaciones dadas al ingresar valores a la variable “y”, por esta razón al hacer el análisis podemos observar la aparición de muchas estructuras de control condicionales.

**PROGRAMA 3: TRIANGULOS**

INICIO

Varaible a, b, c: Entero

ESCRIBIR “Por favor, ingresa los valores de las variables a,b y c”

LEER: a, b, c

SI a = b and b = c ENTONCES

ESCRIBIR: “Es un triángulo Equilátero”

FIN SI

DE LO CONTRARIO

SI a = b or a = c or b = c ENTONCES

ESCRIBIR: “Es un triángulo Isósceles”

FIN SI

DE LO CONTRARIO

ESCRIBIR: “Es un triángulo Escaleno”

FIN DE LO CONTRARIO

FIN DE LO CONTRARIO

FIN

En este programa, al igual que en los anteriores, podemos observar el uso de las estructuras de control condicionales para poder llevar a cabo la ejecución del programa,

**PROGRAMA 4: SUMA DE 3 NUMEROS**

INICIO

Variable a, b, c: Entero

ESCRIBIR: “Introduce el valor de las tres variables”

LEER: a, b, c

SI a+b = c ENTONCES

ESCRIBIR: “La suma de los dos primeros números es igual a el tercer numero”

FIN SI

DE LO CONTRARIO

ESCRIBIR: “La suma de los dos primeros números no es igual al tercero”

FIN DE LO CONTRARIO

FIN

De la misma forma, en este último programa debemos utilizar las estructuras de control condicionales para poder cumplir con lo que se pide.

CONCLUSIONES

A diferencia de las primeras prácticas, en estas últimas tres me he sentido un poco más atraído y emocionado al momento de realizar los reportes ya que estas prácticas son esenciales para que en la programación se pueda poner en práctica lo aprendido y de esa manera, realizar trabajos o programas de buena calidad y que cumplan con los requisitos que me están pidiendo.