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

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Ing. Rodríguez Espino Claudia |
| *Asignatura:* | Fundamentos de programación |
| *Grupo:* | 1122, Bloque: 139 |
| *No de Práctica(s):* | Cuarta práctica |
| *Integrante(s):* | Fonseca Ramírez Gadiel Moisés |
|  |  |
|  |  |
| *Semestre:* | 2018-1 |
| *Fecha de entrega:* | 08/Septiembre/2017 |
| *Observaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

**Guía práctica de estudio 04: Diagramas de flujo.**

**Objetivo:**

Elaborar diagramas de flujo que representen soluciones algorítmicas vistas como una serie de acciones que comprendan un proceso.

**Actividades:**

* Elaborar un diagrama de flujo que represente la solución algorítmica de un problema, en el cual requiera el uso de la estructura de control condicional.
* Elaborar la representación gráfica de la solución de un problema, a través de un diagrama de flujo, en el cual requiera el uso de la estructura de control iterativa.

**Actividades realizadas en clase:**

* Primeramente, puesto que este tema ya se había visto en la clase de teoría, fuimos leyendo la práctica enfocándonos en cómo se debe usar la simbología y cómo se debe ordenar el algoritmo correctamente. Como lo puede ser la forma en la que se deben dibujar las líneas de flujo, además de que a cada símbolo sólo le puede llegar una línea de flujo.
* La profesora explicó las estructuras de control, las cuales sirven para ejecutar una condición o la repetición de un conjunto de instrucciones. Estas estructuras se dividen en:
  1. Secuencial

En esta estructura las instrucciones se realizan en el orden en que están escritas.

c = a+b

c

a, b

* 1. Condicional

IF: En esta evalúa una expresión lógica para saber si es verdadera o falsa, en caso de ser falsa, se va al fin del programa, y en caso de ser verdadera, realiza la instrucción indicada.

IF-ELSE: A diferencia de IF, en caso de ser falsa o verdadera, en ambos casos se realiza un proceso.

IF-ELSE

IF

* 1. Repetitiva:

Estas permiten repetir una serie de instrucciones mientras se cumpla una condición, hasta que deje de cumplirse.

Esta se pude hacer de 2 formas:

* 1. Selección de caso

Esta estructura funciona como un “IF”, pero a diferencia de este, la estructura de “selección de caso” permite realizar distintos procesos dependiendo del caso que se tenga, además se le pueden unir otras “selecciones de caso”.

* También se vio que para resolver problemas que son muy complejos se usan las funciones, las cuales son subprocesos que juntos forman una solución para el problema. Las funciones podrían ser por ejemplo, las operaciones básicas (suma, resta, multiplicación y división), en una función para resolver operaciones complicadas. Estas funciones dentro del código de un programa, es como si el programa les pidiera un dato de alguna operación y las funciones se lo otorgaran.
* Al final se dejó el anexar a la práctica el algoritmo y diagrama de flujo de los siguientes programas:
  1. “La chicharronera”. Ecuación general de segundo grado

Algoritmo:

1. Inicio.
2. “El valor de ’a’ no debe ser cero”, pedir el valor de ‘a’.
3. Si a=0, enviar mensaje “Error, ‘a’ debe ser distinto de cero”, y regresar al paso 1; de lo contrario, seguir al paso 3.
4. Pedir el valor para ‘b’ y ‘c’.
5. Realizar la operación d=b2-4ac.
6. Si d<0 entonces realizar las operaciones: e imprimir x1=r+n’i’ y x2=r-n’i’. En caso contrario ir al paso 6.
7. Realizar las operaciones: y e imprimir ‘x1’ y ‘x2’.
8. Fin.

Diagrama de flujo

d < 0

Error, ‘a’ debe ser distinto de cero.

Sí

No

b, c

a = 0

a

El valor de ‘a’ no debe ser cero.

Sí

No

* 1. Pedir tres valores al usuario y decir si se puede armar un triángulo equilátero, isósceles o escaleno.

-Una de las propiedades de existencia de triángulos es que la suma de dos de sus lados debe ser siempre mayor que el otro.

Algoritmo:

1. Inicio.
2. Imprimir “Escribe el valor de los lados de un triángulo”. Pedir a, b, c.
3. Si a, b, c =0. Imprimir “Error. Ningún lado de un triángulo puede medir cero” e ir al paso 2. En caso contrario ir al paso 4.
4. Si a+b>c o b+c>a o c+a>b, ir al paso 5. En caso contrario imprimir “Estos valores no forman un triángulo” e ir al paso 7.
5. Si a=b y b=c, imprimir “Es un triángulo equilátero”. Si no ir al paso 6
6. Comparar si

a=b, c ≠ a, b

b=c, a ≠ b, c

c=a, b ≠ c, a

Si sí, imprimir “Es un triángulo isósceles”. Si no, imprimir “Es un triángulo escaleno”.

1. Fin.

a, b, c

Escribe el valor de los lados de un triángulo

Error. Ningún lado de un triángulo puede medir cero.

a, b, ó c = 0

Sí

No

Es un triángulo isósceles

Es un triángulo equilátero

Es un triángulo escaleno

b=c, a ≠ b, c

a=c, b ≠ a, c

a=b, c ≠ a, b

a, b, c

a≠b≠c

a=b=c

a, b, c

* 1. Pedir tres valores al usuario y decir si la suma de los 2 primeros es igual al tercer número.

Algoritmo

1. Inicio.
2. Imprimir “Escribe 3 valores”. Pedir ‘a’, ‘b’ y ‘c’. //’c’ es el último valor.
3. Hacer a+b=c. Si a+b=c, imprimir “La suma de los dos primeros valores es igual al tercer número” e ir al paso 4. En caso contrario, ir al paso 3
4. Imprimir “La suma de los dos primeros valores es diferente del último número”.
5. Fin.

Diagrama de flujo

Escribe 3 valores.

c = a + b

a, b, c

La suma de los dos primeros valores es igual al tercer número

Sí

c=a+b

No

La suma de los dos primeros valores es diferente del último número

**Conclusión**

Está práctica permitió aprender acerca del cómo se deben elaborar de manera correcta los diagramas de flujo, para así poder usarlos como u método que represente gráficamente el proceso de solución a un problema. Tomando en cuenta las situaciones en las que se tienen que usar las “funciones”, para que así cuando se tenga que resolver un problema cuya solución tiene un alto nivel de complejidad, estas sirvan para crear las partes de la solución final del problema, manteniendo orden en el diagrama de flujo. Además de que se aprendió de la selección de caso, que ayuda a comparar varias condiciones para un mismo grupo de datos.