

Instituto Tecnológico de Culiacán

Ingeniería en Sistemas Computacionales
UNIDAD IV

Fundamentos de Programación

Mtra. María Lourdes Armenta

Octubre de 2014

Unidad 4. Control de Flujo

- 4.1 Estructuras Secuenciales
- 4.2 Estructuras Repetitivas: simple, doble y múltiple.
- 4.3 Estructuras Iterativas: repetir, mientras, hasta, desde
- 3.4 Diseño e Implementación de funciones

Resumen Estructuras selectivas

- Sentencia if

- Una alternativa:

- If (a !=0)

- resultado = a/b;

- Dos opciones:

- If (a >=0)

- System.out.println(a+" es positivo);

- Else

- System.out.println(a+" es negativo);

- Múltiples opciones

Resumen Estructuras selectivas

- Sentencia if

- Múltiples opciones

- If ($x < 0$)

- { System.out.println("Negativo");

- abs_x=-x;

- else if ($x == 0$)

- { System.out.println("Cero");

- abs_x=0;

- else

- If ($x < 0$)

- { System.out.println("Positivo");

- abs_x=-x;

```
int numeroDia=0;
switch (numeroDia)
{ case 1:
    System.out.println("Domingo"); break;
  case 2:
    System.out.println("Lunes") break;
  case 3:
    System.out.println("Martes"); break;
  case 4:
    System.out.println("Miercoles"); break;
  case 5:
    System.out.println("Jueves"); break;
  case 6:
    System.out.println("Viernes"); break;
  case 7:
    System.out.println("Sabados"); break;
  default:
    System.out.println("Incorrecto");}
```

Resumen estructuras iterativas

- En programación es habitual tener que repetir la operación de una sentencia, lo cual puede realizarse por medio de un bucle; este último es un grupo de instrucciones que se ejecutan reiteradamente hasta que se cumple una condición de terminación. Los bucles representan estructuras de control repetitivas y su número puede establecerse inicialmente o depender de la condición verdadera o falsa.

- Diseño de bucles: 1) repetición controlada por contador y 2) repetición controlada por condición.
- Una variable de control del bucle se utiliza para contar las repeticiones de un grupo de sentencias; se incrementa o decrementa normalmente en 1 cada vez que el grupo se ejecuta.

- La condición de finalización de bucle se utiliza para controlar la repetición cuando las iteraciones no se conocen por adelantado; un valor centinela se introduce para establecer el hecho que determinará si se cumple o no la condición

Bucle *for*

- Inicializa una variable de control a un valor y enseguida comprueba una expresión; si es verdadera, se ejecutan las sentencias del cuerpo del bucle.
- La expresión se comprueba cada vez que termina la iteración; cuando es falsa, se termina y la ejecución sigue en la siguiente sentencia. Es importante que en el cuerpo del bucle haya una sentencia que haga que alguna vez sea falsa la expresión que controla la iteración del bucle.

Bucle *While*

- Comprueba una condición; si es verdadera, se ejecuta la sentencias del bucle. Después se vuelve a comprobar la condición; si sigue siendo verdadera, se ejecutan las sentencias del bucle. Esto finaliza cuando la condición es falsa. La condición esta en la cabecera del bucle, por eso el número de veces que se repite puede ser entre 0 y n.

Bucles *Do...While*

- También comprueban una condición; se diferencian de *while* en que comprueban la condición al final, en vez de la cabecera.

Sentencia *break* y *continue*.

- La sentencia *break* produce la salida inmediata del bucle.
- La sentencia *continue* se ejecuta todas las sentencias siguientes se saltan y, si se cumple la condición de bucle, comienza a una nueva iteración.