



fanzine de programación
☐ valor \$ ☐
by *Acaca*
05/04/2015

HOLA MUNDO

cuaderno de programación
mi nombre



```
int[,] datos;  
datos = new int[3,4];  
datos[0,0] = 5;
```

★ es posible declarar e inicializar un vector o matrix así:

```
int[] x = new int[] {1,2,3,4};  
int[] y = {3,5,6,2,-1};  
int[,] m = new int[2,2] {{2,3},{1,9}};
```

ASIGNA
UN VALOR
A LA CELDA



relación: *mum!*

> == < >= != <=

mayor igual menor mayor o igual diferente menor o igual

evalúan una relación entre dos números y nos dice si es verdadera o falsa

5 > 2

¡sí! 5 es mayor que 2

24

09

ANEXO

estructuras de programación

switch: es como tener varios if anidados.

```
switch(opcion){  
  case 1: //código 1  
    break;  
  case 2: //código 2  
    break;  
  default: //Opción incorrecta  
}
```

operador ternario: es un if abreviado.

```
variable = (condición)?valor1:valor2;
```

```
foreach(i in array){  
  if(5>3){  
    break; // solo del ciclo  
  }  
}
```

break: palabra reservada para salir de un ciclo o de otra estructura.

30

VARIABLE

una variable es un contenedor vacío similar a un vaso, el cual tiene una capacidad

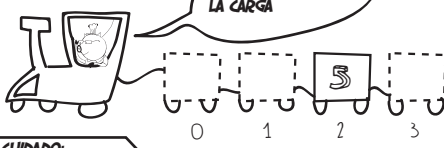


03

22

```
datos[2] = 5;
```

SE EMPIEZA A LLENAR LA CARGA



CUIDADO: SI PONE CARGA EN UNA POSICIÓN QUE NO EXISTE, SU CARGA SE PIERDE.



-1 X error

5 X error

lógicos:



detrás de alguna de estas puertas está la mujer de su vida



&&

and

||

or

!

negación

&&

retorna true si los dos valores son verdaderos, falso en caso contrario

||

retorna false si los dos valores son falsos, verdadero en caso contrario

!

retorna lo contrario, true si el valor es falso o falso si el valor es verdadero

11

FUNCIONES

son módulos de código reutilizable. cuando se tiene un código que se usa varias veces en el programa, se puede crear una función para reusar su comportamiento.

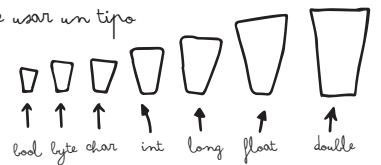
a b → → c

```
a = 5;  
b = 6;  
c = sumar(a,b);
```

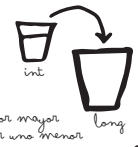
x → → y = sen x

```
x = 45;  
y = seno(x);
```

para crear una variable se debe usar un tipo



el tipo determina cuanto espacio se requiere para representar el dato. por ejemplo si hablamos de números, los tipos de dato son como las matrices

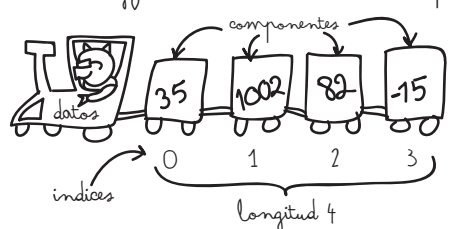


un contenedor mayor puede guardar uno menor

05

VECTORES

es una estructura de programación que agrupa un número fijo de variables de un mismo tipo



20



centinela: vigila y espera a que un valor se produzca.

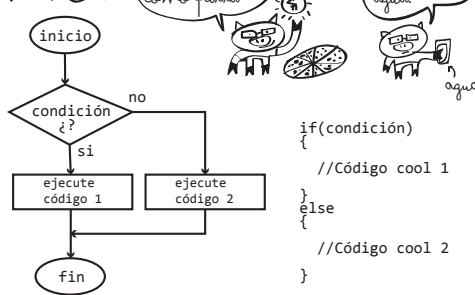
```
int centinela = 0;  
do {  
  //procedimiento para ingresar  
  //un valor en valor_ingresado  
  centinela= valor_ingresado;  
}while(centinela != -1);
```

vigila que el valor ingresado sea -1 apenas ocurre el ciclo termina.

07

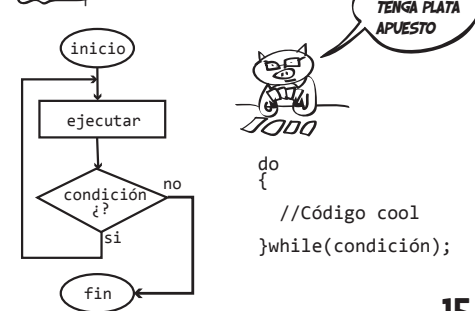
18

si-sino:



13

hasta que:



15

s[0] → devuelve un char, que es la representación del carácter en la posición especificada por el índice en este caso 0.

s[0] → h

las cadenas son inmutables, pero aun así tiene poderes los cuales llamaremos comandos.



26

por el amor a la vida, no confunda

b=5;
asigna a b el valor de 5

= con **==**
operador de asignación operador de relación de igualdad

ud **==** ud
¿ud es igual a ud mismo?
si (true)

10

ESTRUCTURAS DE CONTROL

si:

```

    inicio
    ↓
    condición ?
    /  \
  si   no
  ↓    ↓
ejecute algo
  ↓
    fin
  
```

si tengo la llave entro

```

if(condición)
{
    //Código cool acá
}
  
```

12

cielos

mientras que:

```

    inicio
    ↓
    condición ?
    /  \
  si   no
  ↓    ↓
ejecutar
  ↓
    fin
  
```

SI NO ESTOY CANSADA, CAMINO.

```

while(condición)
{
    //Código cool
}
  
```

14

para:

```

    inicio
    ↓
    i = 0
    ↓
    i < 10
    /  \
  si   no
  ↓    ↓
ejecutar
  ↓    ↓
i = i + 1
  ↓
    fin
  
```

inicialización

UN DISPARO PARA CADA DIANA

```

for(i=0; i<10; i++)
{
    //Código cool
}
  
```

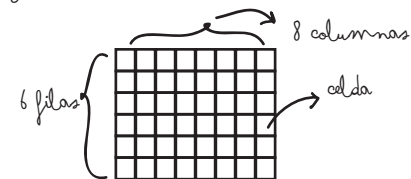
condición

incremento o decremento

16

MATRICES

es una estructura que permite almacenar datos del mismo tipo en forma de tabla, con N filas y M columnas.



```
int[] datos;
```

SE CREA EL TREN

EL TREN SIN VAGONES NO ES UTIL

```
datos = new int[4];
```

SE RESERVA ESPACIO PARA CUATRO VAGONES

0 1 2 3

21

contador: cuenta valor, por lo general de uno en uno.

```

int contador = 0;
while(contador < 100)
{
    contador = contador + 1;
}
  
```

acumulador: es similar a un contador, pero acumula en multiples, sumando, dividiendo, multiplicando, entre otros.

```

int suma = 2;
while(suma < 100)
{
    suma = suma * 2;
}
  
```

19

BESTIARIO DE VARIABLES

bandera: es una llave, que controla como se ejecuta el flujo del programa.

```

bool bandera = true;
while(bandera)
{
    //salga de acá
    bandera = false;
}
  
```

17

TABLA DE CONTENIDO

3 variables	20 vectores
8 operadores	23 matrices
12 estructuras de control	25 cadenas
17 bestiario de variables	28 funciones y tipos

23 02

en programación antes de usar una variable, es necesario, realizar dos operaciones:

inicializar la variable
es poner contenido en el vaso, para poder usar el valor

declarar la variable
(es conseguir el vaso)
reservar el espacio donde se almacenará el dato.

21 04

21 04

muy importante

si el contenido a guardar en la variable supera su capacidad, hay un desbordamiento.

si queremos guardar el contenido de un vaso mayor, en un contenedor menor, no va a ser posible porque hay un desbordamiento o pérdida de datos.

vaso cine

vaso café

19 06

19 06

OPERADORES

aritméticos

suma

resta

multiplicación

división

úselos, son gratis lo van a salvar

orden de evaluación

1. () paréntesis
2. * y /
3. + y -

se evalúan de izquierda a derecha

paréntesis (Heroe)

17 08

17 08

MIS BONIFICACIONES

peque aquí su bonificación

tipos de funciones:

no recibe datos de entrada y no retorna datos de salida

saludar()

recibe datos de entrada sin retorno de datos de salida

guardar(325);

con datos de entrada y con retorno

total = sumar(3,5);

31

algunos comandos:

"hola" Equals false

"hola mundo"

"m" IndexOf 5

"hola mundo"

"hola" Replace "adios mundo"

para consultar el resto de comandos, consulte en el api del lenguaje en internet

27

CADENAS

una cadena es un dato alfanumérico o un conjunto de caracteres, funciona similar a un vector.

```
string s = "hola mundo";
```

0 1 2 3 4 5 6 7 8 9

s.Length

10, total de caracteres

25