

EasyCODE(C++) Demo - [alterna.c * - Top:Leer respuesta y escribir sí o no

File Edit Insert View Add-In Options Window ?

Leer respuesta y escribir sí o no

```
#include <stdio.h>

function main,
    char inicial;

    printf("Dar respuesta (s ó n): ");
    scanf("%c",&inicial);

    if ( inicial == 's' )
        then
            printf("Has dicho sí\n");
        else
            printf("Has dicho no\n");
```

<http://www.easycode.de/home.html?&L=1>

Ready ● W x H 1 x 1 NUM



Objetivos

- Presentar el entorno de diseño, programación y ejecución EasyCode.
- Describir los elementos de EasyCode.



Introducción

- EasyCODE es un producto de tipo *ingeniería de software asistido por ordenador*, que facilita la tarea de diseño de programación, utilizando técnicas de *programación estructurada*.
- Con EasyCODE se puede crear, editar e imprimir algoritmos en la forma de diagramas estructurados.
- La versión a utilizar es EasyCode 6.8, que se puede descargar en:

http://personales.unican.es/corcuerp/EasyCode/easycode_68_cpp_en.zip



Programación estructurada

- Método para construir algoritmos a partir de un número pequeño de bloques básicos.
- Formas fundamentales:
 - **Secuencia:** indica secuencia temporal lineal de las acciones a realizarse.

A

B



Programación estructurada

- **Selección:** especifica una condición que determina la acción a realizarse.

`if` C

D

`else`

E

- **Repetición:** indica que una o más acciones deben repetirse un determinado número de veces.

`while` G

H



Programación estructurada

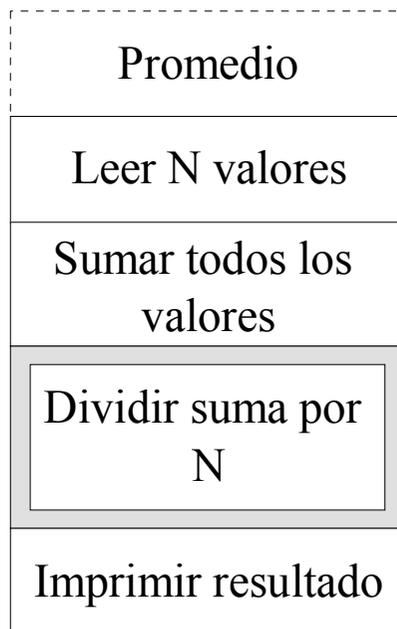
- **Invocación:** corresponde al grupo de acciones agrupadas bajo un nombre.
 Calcula_Promedio



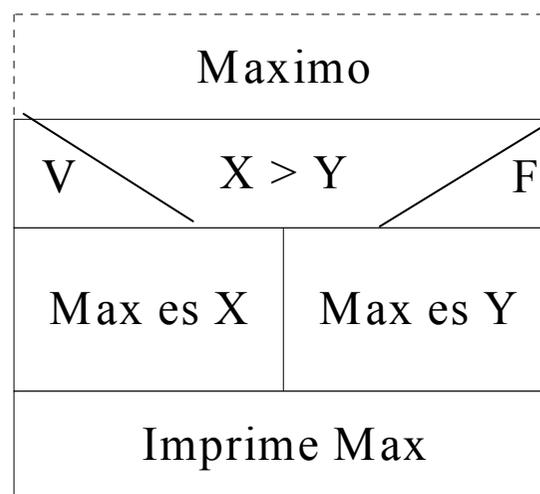
Diagramas de bloques o NS

- Básicamente se trata de cajas rectangulares que se apilan de acuerdo al algoritmo.

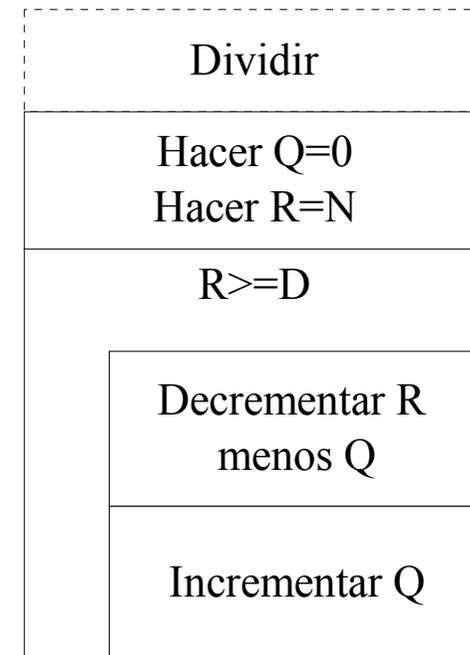
Secuencia



Selección

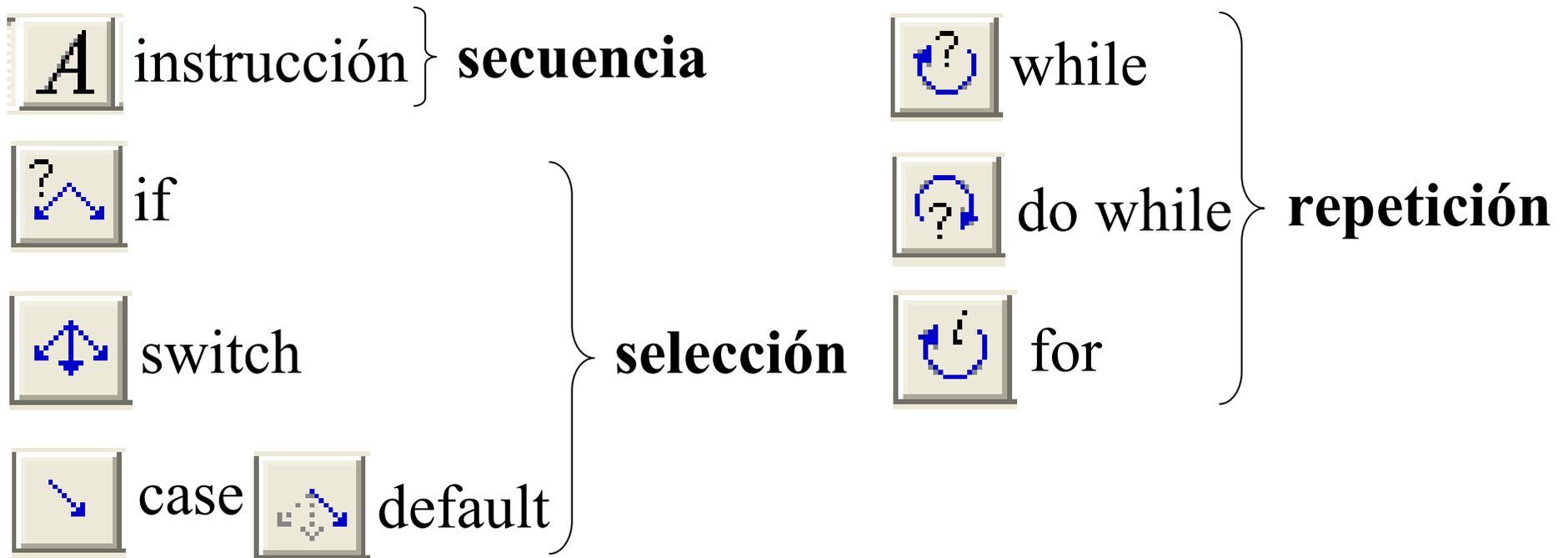
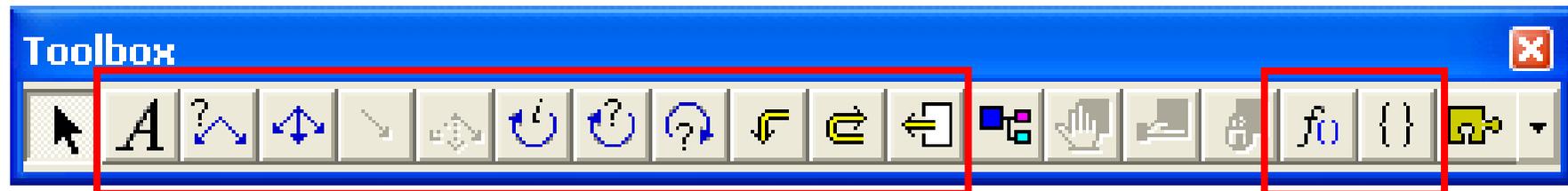


Repetición



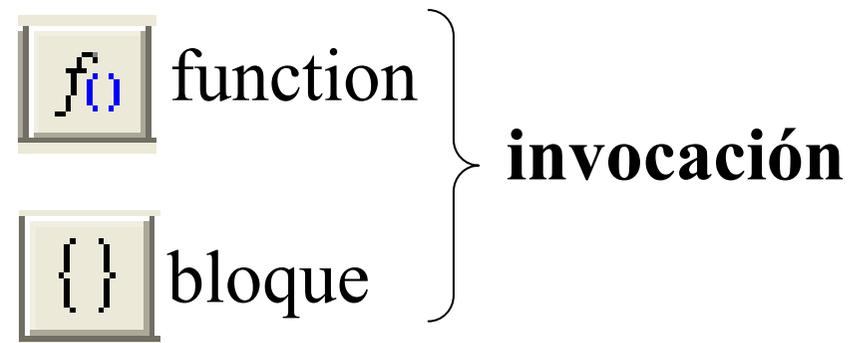
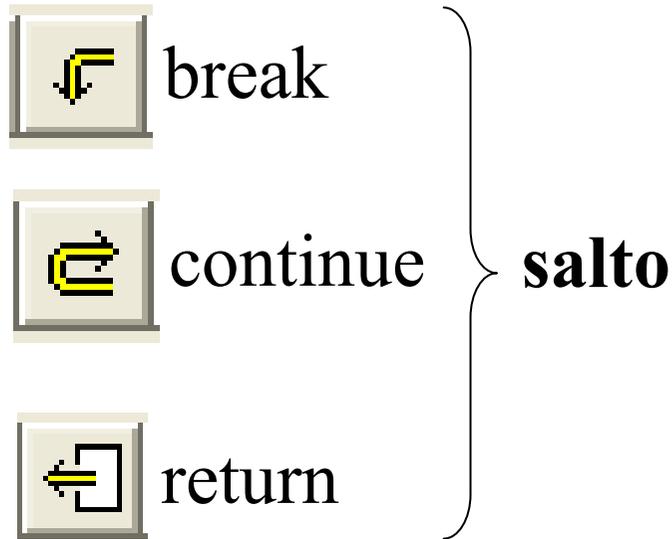


EasyCode: Barra de herramientas





EasyCode: Barra de herramientas





Ejemplo

- Elaborar un algoritmo en EasyCode que pregunte al usuario un número entero positivo **n**, menor a 1000, y que calcule el producto de 1 a n.

$$prod = \prod_{i=1}^n i$$

- Guardar el fichero como **factorial.c**



Ejemplo

- Pseudocódigo:

Input n

prod = 1

i = 2

while i <= n

 prod = prod * i

 i = i + 1

endwhile

Output prod

- Con EasyCode:

Programa para calcular el factorial de N

Input N
prod = 1
i = 2
while (i <= N)
prod = prod * i
i = i + 1
Output prod