

ejemplo de función acertar un numero

//Pide un numero al usuario

Funcion numeroUsuario <- leerNumero ()

Repetir

 Escribir "Dame un numero entre 1 y 100"

 leer numeroUsuario

 Hasta Que numeroUsuario>=1 Y numeroUsuario<=100

Fin Funcion

//Comprueba el numero y devuelve un codigo

//0: los dos numeros son iguales

//1: el numeroUsuario es mayor que el numeroCorrecto

//-1: el numeroUsuario es menor que el numeroCorrecto

Funcion codigo <- comprobarNumero (numeroCorrecto, numeroUsuario)

 Si numeroUsuario = numeroCorrecto Entonces

 codigo <- 0

 Sino

 Si numeroUsuario < numeroCorrecto Entonces

 codigo <- -1

 Sino

 codigo <- 1

 Fin Si

Fin Si

Fin Funcion

Algoritmo Ejercicio_DDR_subprocesos_9

 numeroCorrecto <- 43

 salir <- Falso

 //Hasta que no acierte no salimos

 Mientras no salir Hacer

 //leer el numero del usuario

 numeroUsuario <- leerNumero()

 //Comprobar el numero del usuario

 Segun comprobarNumero(numeroCorrecto, numeroUsuario) Hacer

 0:

 //Salimos

 Escribir "Has acertado, felicidades"

 salir <- verdadero

 1:

 Escribir "El numero que has introducido es mayor que el numero buscado"

 -1:

 Escribir "El numero que has introducido es menor que el numero buscado"

 Fin Segun

 Fin Mientras

FinAlgoritmo

segundo ejemplo de funcion

```
Funcion resultado <- Maximo( n1, n2)
```

```
  Si n1 > n2 Entonces
```

```
    resultado <- n1
```

```
  Sino
```

```
    resultado <- n2
```

```
  FinSi
```

```
FinFuncion
```

```
Algoritmo Funciones01
```

```
  Escribir "El máximo de 5 y 12 es:"
```

```
  Escribir Maximo(5,12)
```

```
  Escribir "El máximo de 25 y 12 es:"
```

```
  Escribir Maximo(25,12)
```

```
FinAlgoritmo
```

ejemplo de proceso

```
SubProceso cod<-pir_inver(altura)
```

```
  numast<-3+2*(altura-2)
```

```
  nespacios<-numast
```

```
  espacios=""
```

```
  si altura=0 Entonces
```

```
    cod<- (-1)
```

```
  Sino
```

```
    Para i<-numast hasta 1 con paso -2
```

```
      numast<-i
```

```
      para k<-0 hasta numast-nespacios Con Paso 1
```

```
        espacios<-espacios+" "
```

```
      FinPara
```

```
      Escribir Sin Saltar espacios
```

```
      para j<-1 hasta numast con paso 1
```

```
        Escribir Sin Saltar "*"
```

```
      FinPara
```

```
      nespacios<-nespacios-2
```

```
      Escribir ""
```

```
    FinPara
```

```
    cod<-0
```

```
  FinSi
```

```
FinSubProceso
```

```
Proceso ejercicio
```

```
  Escribir "Introduce la altura de la pirámide invertida"
```

```
  leer altura
```

```
  resultado<-pir_inver(altura)
```

```
  si resultado=-1 entonces
```

```
    Escribir "ha introducido una altura 0"
```

```
  Sino
```

```
    Escribir ""
```

```
    Escribir "introduccion de datos correcta"
```

```
  FinSi
```

FinProceso

segundo ejemplo de proceso

SubProceso suma<- fibonacci (posicion)

 num1<-0

 num2<-1

 suma<-1

 contador<-1

 Mientras (contador<posicion)

 suma<-num1+num2

 num1<-num2

 num2<-suma

 contador<-contador+1

 FinMientras

FinSubProceso

Proceso ejercicio_funciones_2

 Leer posicion

 resultado<-fibonacci(posicion)

 Escribir resultado

FinProceso