

# SilverTech

+talento +tecnología +empleabilidad

**Socio estratégico**

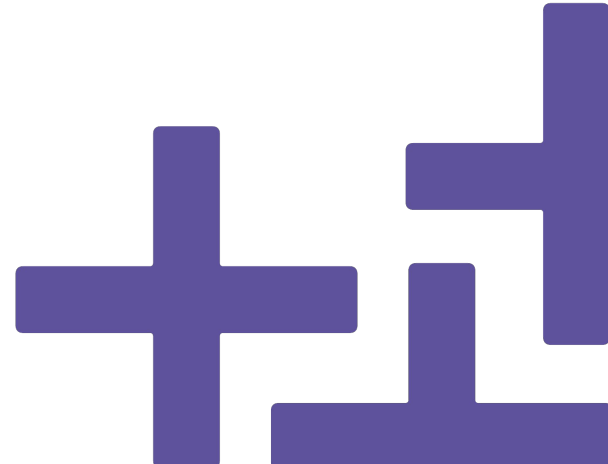


**Impulsan**



# Cobol – Clase 25

Rutinas SubPrograma



# Reglas de la clase



Micrófonos apagados



Consultas al final de la clase



Consultas por chat



# Cronograma

## Primera Parte

18:30  
a  
19:25

## Break

19:25  
a  
19:35

## Segunda Parte

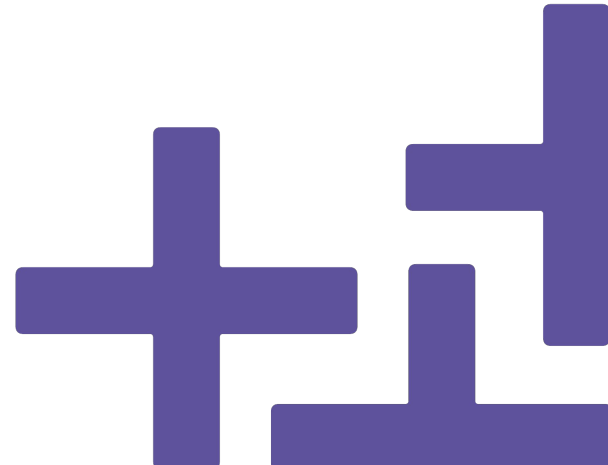
19:35  
a  
20:30

# ¿Qué veremos hoy?

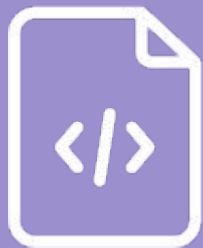
+ Modularización

+ Subprogramas.

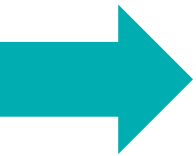
+ Ejercicios.



# Modularización



Consiste en dividir una función principal en funciones por separado. Con el objetivo de descomponer un proceso grande o complejo en pequeños procesos más simples y con objetivos más concretos.

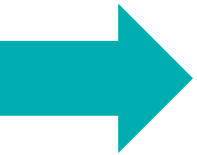


# ESTRUCTURA MODULAR

- + La Implementación de la **ESTRUCTURA MODULAR** se puede llevar a cabo a través de dos sentencias:

PERFORM	CALL
<p>OBJETO → Párrafo</p> <p>▼</p> <p>Dentro del mismo programa</p>	<p>OBJETO → Subprograma</p> <p>▼</p> <p>Fuera del programa principal (en una biblioteca)</p>

- + Del mismo modo que el PERFORM al finalizar su ejecución la instrucción CALL vuelve a la instrucción siguiente.



# USO DE SUBPROGRAMA

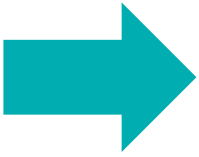


La tarea es demasiado grande para ser realizada por un solo programa, de esta forma el Subprograma sería una forma natural de distribuir las tareas entre varias.

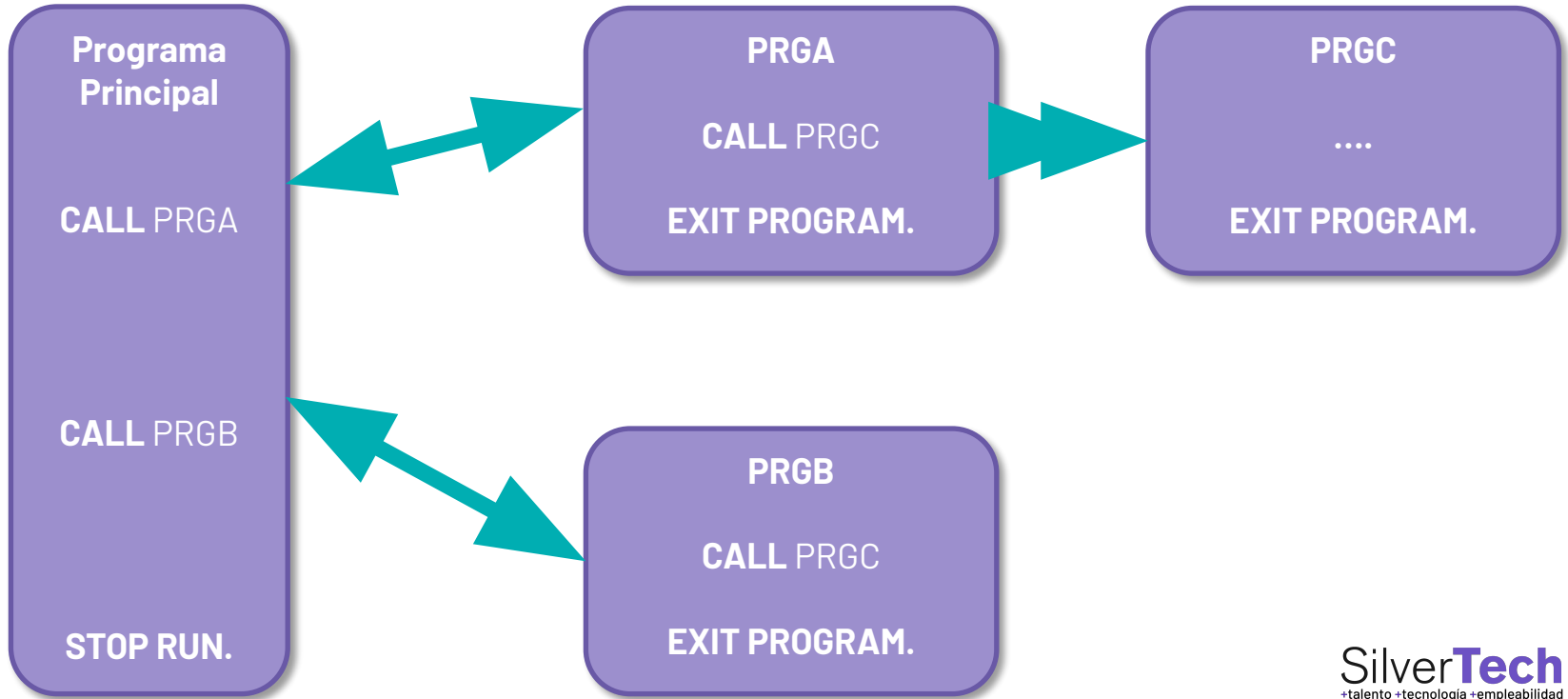


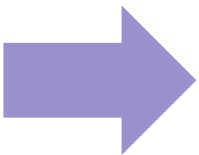
La misma tarea necesita ser incorporada a varios programas.





# COMPILADO





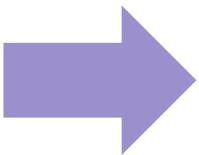
# TIPO DE LLAMADOS

01.

Call

02.

Link



# CARACTERÍSTICAS



## CALL

- Más eficientes.
- Solo pueden usarse en regiones LOCALES.
- El módulo llamado es parte del módulo de carga.
- El sistema operativo no interviene en el llamado.

## LINK

- Más fácil de debugear.
- Pueden usarse en diferentes regiones.
- El programa llamador cede el control al programa llamado.
- El sistema operativo interviene en el llamado.

# Instrucción

**CALL** **identificador** **USING** [nombre-dato-2 ... nombre-dato-n]  
[**ON OVERFLOW** sentencia-1]



- **Identificador-1**: nombre del subprograma al que se llama.



- **Identificador-2**: nombre de la variable que recibirá los datos de salida (opcional).

- **ON OVERFLOW**, indica una condición de error, tal como imposibilidad de acceso al subprograma, memoria insuficiente y similares.

# Instrucción

**EXEC CICS LINK PROGRAM**

**PROGRAM (identificador)**

**COMMAREA (nombre-dato)**

**RETCODE(código-estado)**

**LENGTH(tamaño)**

**END-EXEC**

- **Identificador**: nombre del subprograma al que se llama.
- **nombre-dato** : Área de entrada/salida .
- **código-estado** : Variable para devolver código de estado numérico **PIC S9(8) comp.**
- **tamaño** : Tamaño de área de comunicación. 32K




# Llamado LINK cics

```
001505      EXEC CICS
001506          LINK PROGRAM ('CLVALCBU')
001507              COMMAREA (NCBU-AREA)
001508              LENGTH (4096)
001509              RESP      (WS-RESP)
001510      END-EXEC.
001511
001512      * SE EVALUA PARA QUE NO ABENDEE
001513      EVALUATE WS-RESP
001514          WHEN DFHRESP(NORMAL)
001515              CONTINUE
001516          WHEN OTHER
001517              MOVE 'VALCBUER'          TO LOGT-ID-ERROR-I
001518              MOVE '90'                TO WS-COMO-RETORNO
001519              PERFORM 9100-RETORNAR-ER
001520                  THRU 9100-RETORNAR-ER-EXIT
001521      END-EVALUATE.
```



# Tipo de Link



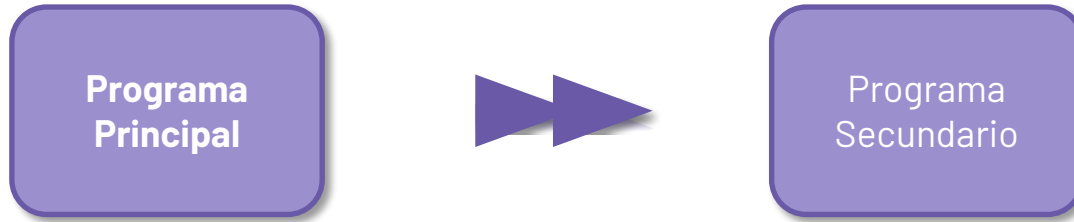
**Llamada Estática**  se produce cuando un programa es compilado con la opción del compilador NODYNAM. Un estático llamado programa se carga en el almacenamiento en tiempo de compilación.



**Llamada Dinámica** se produce cuando se compila un programa con la DYNAM y NODLL opción del compilador. Un dinámico programa llamado se carga en el almacenamiento en tiempo de ejecución.

# Subprograma

- Un Programa Principal puede llamar a uno o más Subprogramas.



## PROGRAMA PRINCIPAL:

- Es el Ejecutable
- Inicia y controla la ejecución de los Subprogramas y la finalización de la tarea.

## SUBPROGRAMA:

- Se ejecuta sólo con el Programa Principal.
- Se escribe y se compila independientemente.
- Puede ser llamado por el Programa Principal o por otro Subprograma pero nunca por sí mismo.





# ASPECTOS DEL SUBPROGRAMA Y DEL PROGRAMA PRINCIPAL

SUBPROGRAMA	PROGRAMA
<b>PROGRAM-ID.</b> VALIDACUIT. ... <b>LINKAGE SECTION.</b> 01 LK-ENTRADA. 05 LK-MENSAJE-ERROR    PIC X(20). 05 LK-CUIT              PIC 9(11). 05 LK-CODIGO-ERROR      PIC 9(2).  ... <b>PROCEDURE DIVISION USING</b> MENSAJE-ERROR CUIT CODIGO-ERROR.  ... <b>.EXIT PROGRAM.</b>	<b>PROGRAM-ID.</b> PROVEEDORES. ... <b>WORKING-STORAGE SECTION.</b> 01 WS-ENTRADA. 05 WS-MENSAJE-ERROR    PIC X(20). 05 WS-CUIT              PIC 9(11). 05 WS-CODIGO-ERROR      PIC 9(2).  . . <b>PROCEDURE DIVISION.</b> ... <b>CALL 'VALIDACUIT' USING</b> MENSAJE NROCUIT TIPO-ERROR.  <b>STOP RUN.</b>



# COMMAREA DE ENTRADA Y SALIDA

Podemos definir una commarea para enviar/recibir datos de un subprograma

- + Podemos definir una porción fija de área para asignar a la commarea de entrada.

Y otra porción fija de área para asignar a la commarea de salida.

- + Podemos definir una porción fija de área para asignar a la commarea de entrada, y luego redefinir la misma área para asignarsela a la commarea de salida.



# Definición de commarea

```
000823      LINKAGE SECTION.
000824      01 DFHCOMMAREA.
000825      *01 LK-AREA.
000826      *AREA DE INPUT
000827          03 LK-INPUT.
000828              05 LK-FUNCION-I                      PIC X(01) .
000829              05 LK-SUBFUNC-I                      PIC X(02) .
000830              05 LK-FILLER-I                      PIC X(2015) .
000831      *FUNCION 'C' SUBFUNCION '01'
000832          05 LK-C01-I REDEFINES LK-FILLER-I.
000833              07 LK-C01-IDE-USUA-I                  PIC X(35) .
000834              07 LK-C01-COD-MODU-I                  PIC 9(4) .
000835              07 LK-C01-FILLER-I                    PIC X(1976) .
000836      *AREA DE OUTPUT
000837          03 LK-OUTPUT.
000838              05 LK-FILLER-O                      PIC X(2018) .
000839      *FUNCION 'C' SUBFUNCION '01'
000840          05 LK-C01-O REDEFINES LK-FILLER-O.
000841              07 LK-C01-FILA-O OCCURS 40.
000842      *CANCAR = 033
000843              09 LK-C01-COD-SUBS-O                  PIC 9(04) .
000844              09 LK-C01-COD-MONE-O                  PIC X(02) .
000845              09 LK-C01-NRO-CNTA-O                  PIC X(25) .
000846              09 LK-C01-TIP-CNTA-O                  PIC X(02) .
000847              07 LK-C01-FILLER-O                    PIC X(298) .
```

# Definición de commarea - Redefine

```
000823      LINKAGE SECTION.
000824      01 DFHCOMMAREA.
000825      *01 LK-AREA.
000826      *AREA DE INPUT
000827          03 LK-INPUT.
000828              05 LK-FUNCTION-I                PIC X(01) .
000829              05 LK-SUBFUNC-I                PIC X(02) .
000830              05 LK-FILLER-I                PIC X(3994) .
000831      *FUNCION 'C' SUBFUNCION '01'
000832              05 LK-C01-I REDEFINES LK-FILLER-I.
000833              07 LK-C01-IDE-USUA-I          PIC X(35) .
000834              07 LK-C01-COD-MODU-I          PIC 9(4) .
000835              07 LK-C01-FILLER-I            PIC X(3994) .
000836      *AREA DE OUTPUT
000837          03 LK-OUTPUT REDEFINES LK-INPUT.
000838              05 LK-FILLER-O                PIC X(4036) .
000839      *FUNCION 'C' SUBFUNCION '01'
000840              05 LK-C01-O REDEFINES LK-FILLER-O.
000841              07 LK-C01-FILA-O OCCURS 90.
000842      *CANCAR = 033
000843              09 LK-C01-COD-SUBS-O          PIC 9(04) .
000844              09 LK-C01-COD-MONE-O          PIC X(02) .
000845              09 LK-C01-NRO-CNTA-O          PIC X(25) .
000846              09 LK-C01-TIP-CNTA-O          PIC X(02) .
000847              07 LK-C01-FILLER-O            PIC X(166) .
```



# Comunicación

**Foro de consultas TEC:**

<https://campus.soysilverttech.org>

**Mails de consulta TEC:**

[consultasCOBOL@soysilverttech.org](mailto:consultasCOBOL@soysilverttech.org)



# GRACIAS

Silver**Tech**  
+talento +tecnología +empleabilidad



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

SilverTech

+talento +tecnología +empleabilidad