

# RPG

## *Quo vadis?*



Marco Riva

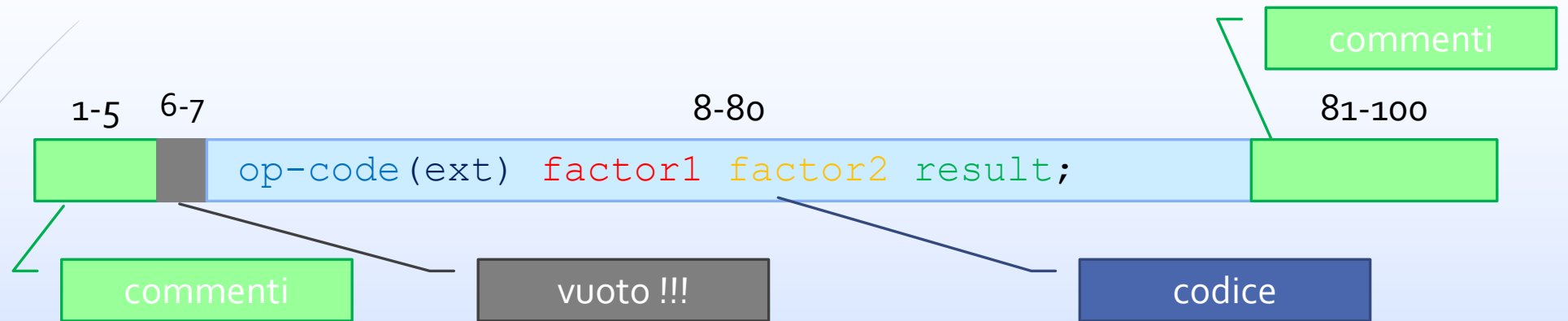


[www.markonetools.it](http://www.markonetools.it)



Ultimo aggiornamento: 24/01/2021

# Regole base



- ogni istruzione deve terminare con il carattere **;**
- `/free ... /end-free` **non** sono più necessari da **7.1 TR7**



# Codici operativi "obsoleti"

- ▶ ATTENZIONE: non tutti i codici operativi possono essere scritti in formato libero: `ADD`, `ANDxx`, `CABxx`, `CALL`, `CALLB`, `CASxx`, `CAT`, `COMP`, `DEFINE`, `DIV`, `DOUxx`, `DOWxx`, `GOTO`, `IFxx`, `MOVE`, `MOVEA`, `MOVEL`, `MULT`, `MVR`, `ORxx`, `PARM`, `PLIST`, `SCAN`, `SETOFF`, `SETON`, `SQRT`, `SUB`, `SUBST`, `TAG`, `TESTB`, `TESTN`, `TESTZ`, `TIME`, `WHENxx`, `Z-ADD`, `Z-SUB`
- ▶ I codici operativi non gestiti dal formato libero possono essere definiti "obsoleti" e "deprecabili"

come controllare il flusso procedurale  
in formato libero?

# Cicli iterativi condizionati



## do *do until*

cicla **fino a che** la condizione **diventa vera**

eseguito **almeno 1 volta**

la condizione viene testata a **fine** ciclo

## do *do while*

cicla **mentre** la condizione **è vera**

la condizione viene testata a **inizio** ciclo

## iter *ripeti ciclo*

ovverosia sposta il controllo

a enddo

## leave *esci dal ciclo*

ovverosia sposta il controllo

**dopo** enddo

enddo *fine ciclo*

## **ciclo infinito**

```
do *on;
```

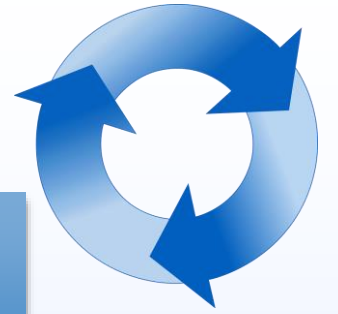
*oppure*

```
dcl-s ForEverTrue ind inz(*on);  
do ForEverTrue;
```

7.3 Sl73190  
7.4 Sl73192  
(mag 2020)

4

# Cicli iterativi enumerati



**for** *index*{=start} {by step} {to|downto limit}  
ripete il ciclo fino a quando la variabile *index* **supera** il valore di *limit*.

**for-each** *item* in *array*|%list|%subarr  
ripete il ciclo per **ogni elemento** dell'array

new 7.4 TR3,  
7.3 TR9

**iter** *ripeti ciclo*  
overrosia sposta il controllo  
**a** endfor

**leave** *esci dal ciclo*  
overrosia sposta il controllo  
**dopo** endfor

**%list**  
restituisce un array  
temporaneo i cui valori  
sono gli argomenti della  
funzione %list

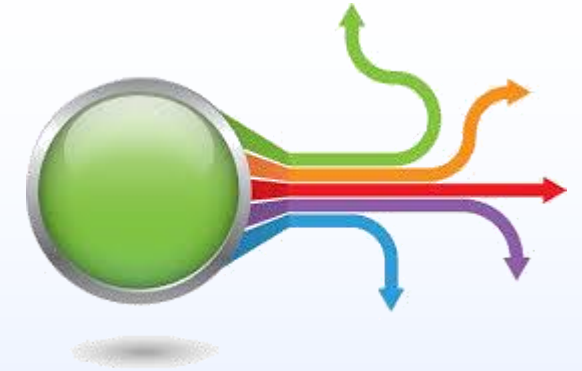
**%subarr**  
restituisce una porzione  
di un array

**ciclo (quasi) infinito**

```
dcl-s i int(10);  
for i = 1;
```

**endfor** *fine ciclo*

# Strutture di controllo



## if, else, elseif, endif

```
if condizionale1;  
    // condizione 1 vera  
elseif condizionale2;  
    // condizione 2 vera  
else;  
    // tutte le condizioni false  
endif;
```

## select, when, other, endsl

```
select;  
    when condizionale 1;  
        // condizione 1 vera  
    when condizionale 2;  
        // condizione 2 vera;  
    other;  
        // tutte le condizioni false  
endsl;
```

x in %list (w : y : z )

è equivalente a

x = w or x = y or x = z

x in %range (y : z)

è equivalente a

x >= y and x <= z



6

# Via di fuga



uscire da una subroutine

## leavesr

```
begsr abc;  
    if condizione;  
        leavesr;  
    endif;  
endsr;
```

nella main procedure  
fare attenzione al  
valore dell'indicatore

**LR**

uscire e tornare al chiamante

## return



7

# Riferimenti



➤ E-mail aziendale: [mriva@sirio-is.it](mailto:mriva@sirio-is.it)



➤ Blog: [www.markonetools.it](http://www.markonetools.it)



➤ E-mail blog: [info@markonetools.it](mailto:info@markonetools.it)



➤ LinkedIn: [www.linkedin.com/in/marcoriva-mk1](http://www.linkedin.com/in/marcoriva-mk1)



➤ Twitter: [@MarcoRiva73](https://twitter.com/MarcoRiva73)



➤ Facebook: <https://www.facebook.com/markonetools/>



➤ YouTube: <https://www.youtube.com/channel/UCb47YJOJCzU-5x4nnGzDu-w>

Power coffee - MK1



8