

La struttura ciclica **do while**

```
do while (condizione logica)
    istruzioni
    istruzioni
end do
```

- Fino a quando è vero il predicato allora verranno eseguite le istruzioni contenute all'interno del ciclo.
- Poi si valuta nuovamente il predicato: se è falso, si esce dal ciclo; invece, se è vero, si eseguono nuovamente le istruzioni del ciclo.
- Si va avanti in questo modo fino a quando il predicato risulta essere falso.

Esempio sul metodo del Punto Fisso

$$x_1 = g(x_0)$$

$$x_2 = g(x_1)$$

$$x_3 = g(x_2)$$

$$\vdots$$

$$x_{k+1} = g(x_k)$$

- Da un punto di vista numerico, supponendo di volere una soluzione approssimata con una certa tolleranza `tol1`, ci si ferma nel calcolo della successione dei valori, quando $|x_{k+1} - x_k| < \text{tol1}$ per un certo valore di k .

Per scrivere un programma FORTRAN

- Parto da un'approssimazione iniziale x_0 .
- Assegno il valore della tolleranza `toll`.
- Pongo $x_k = x_0$.
- Dichiaro la variabile `scarto = |xkp1 - xk|`.
`xkp1` ha il significato dell'iterazione successiva rispetto a `xk`.
- Inizialmente, pongo `scarto` uguale ad un valore più grande della tolleranza `toll`, ad esempio `scarto = 2*toll`.
- Conto il numero delle iterazioni che farò per arrivare a convergenza mediante la variabile `iter`. Al passo iniziale, `iter = 0`.

In linguaggio FORTRAN...

```
do while (scarto.ge.toll)
  iter = iter +1
  xkp1=g(xk)
  scarto = abs(xkp1 - xk)
  xk = xkp1
end do
```

Quando finisce il ciclo?

- Cosa succede se il predicato del `do while` è sempre vero?

Il ciclo durerebbe all'infinito!!!!!!!

- Quindi bisogna prestare molta attenzione sul predicato che si scrive in modo che non possa essere sempre vero!
- **Esempio:** nel calcolo del punto fisso il predicato `scarto > = toll` potrebbe essere sempre vero (ad esempio se non si arriva a convergenza)
- Occorre fissare un numero massimo di iterazioni per il nostro ciclo, in genere un numero grande.

Algoritmo del punto fisso

```
do while ((scarto.ge.toll).and.(iter.le.itmax))
  iter = iter +1
  xkp1=g(xk)
  scarto = abs(xkp1 - xk)
  xk = xkp1
end do
```

- Ora il predicato del ciclo `do while` è:
((scarto.ge.toll).and.(iter.le.itmax)), cioè
deve essere sia `scarto >= toll` e sia `iter <= itmax`
perchè si eseguano le istruzioni contenute all'interno del ciclo
`do while`.
- Il predicato diventa falso quando una delle due proposizioni
logiche non è più vera cioè: `scarto < toll` o `iter >`
`itmax`.

Alla fine del ciclo iterativo

- Se $\text{scarto} < \text{toll}$ allora il metodo è arrivato a convergenza.
- Se invece $\text{iter} > \text{itmax}$ allora il numero delle iterazioni è troppo grande e il metodo o sta divergendo oppure non riesce ad arrivare a convergenza secondo la tolleranza prefissata entro quel numero massimo di iterazioni.

Nota: il motivo esatto lo si capisce dalla stampa delle approssimazioni per ogni iterazione.

Siamo quindi in grado di scrivere un programma sul metodo del punto fisso utilizzando il ciclo `do while`.