

Istruzioni condizionali

Statistica computazionale
Carla Rampichini
a.a. 2008/09

Istruzioni condizionali

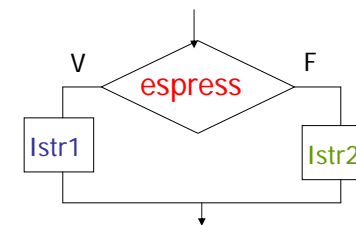
- Subsetting IF
IF espressione;
- IF-THEN-ELSE
IF espress THEN istr;
[ELSE istr;]
- DO WHILE
- DO UNTIL

IF-THEN- ELSE

```
IF espressione THEN istr;  
[ ELSE istr; ]
```

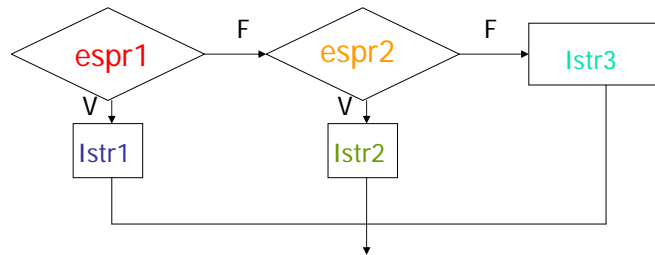
NB espressione \forall !
istr. ESEGUITA SOLO SE
espress. È VERA
NO ϕ , NO.

IF-THEN-ELSE



```
IF eta LT 20 THEN var1=1;  
ELSE var1=2;
```

IF a catena



```
IF eta LT 20 THEN var1=1;
ELSE IF 20<=ETA< 30 THEN var1=2;
ELSE var1=3;
```

IF-THEN-ELSE: più istruzioni

```
IF espr THEN DO;
...
...
END;
ELSE DO;
...
...
END;
```

SUBSETTING IF: comando IF come selezionatore (1)

```
if dim_auto = 'piccola'; lavora solo sulle auto di piccole dimensioni

if 11 <= kpl < 18; lavora solo sulle auto con consumi medi

if paese ne 'giappone'; esclude dall'analisi le auto giapponesi

if (paese = 'usa') and (dim_auto = 'grande'); lavora solo sulle auto americane di grosse dimensioni
```

SUBSETTING IF: comando IF come selezionatore (2)

```
if dim_auto = 'piccola'; lavora solo sulle auto di piccole dimensioni

if 11 <= kpl < 18; lavora solo sulle auto con consumi medi

if paese ne 'giappone'; esclude dall'analisi le auto giapponesi

if (paese = 'usa') and (dim_auto = 'grande'); lavora solo sulle auto americane di grosse dimensioni
```

SUBSETTING IF: comando IF come selezionatore (3)

```
if dim_auto = 'piccola';          lavora solo sulle auto di piccole dimensioni  
  
if 11 <= kpl < 18;                lavora solo sulle auto con consumi medi  
  
if paese ne 'giappone';          esclude dall'analisi le auto giapponesi  
  
if (paese = 'usa') and (dim_auto = 'grande');  lavora solo sulle auto americane di grosse dimensioni
```

SUBSETTING IF: comando IF come selezionatore (4)

```
if dim_auto = 'piccola';          seleziona auto piccole  
  
if 11 <= kpl < 18;                seleziona auto con consumi medi  
  
if paese ne 'giappone';          esclude dall'analisi le auto giapponesi  
  
if (paese = 'usa') and (dim_auto = 'grande');  seleziona auto americane di grosse dimensioni  
  
ne = not equal (≠)
```

SUBSETTING IF: comando IF come selezionatore (5)

```
if dim_auto = 'piccola';          seleziona auto di piccole dimensioni  
  
if 11 <= kpl < 18;                seleziona auto con consumi medi  
  
if paese ne 'giappone';          esclude dall'analisi le auto giapponesi  
  
if (paese = 'usa') and (dim_auto = 'grande');  seleziona auto americane di grosse dimensioni
```

COMANDI IF...THEN...ELSE (1)

```
if kpl >=18 then consumi = 'buono';  
if 11 <= kpl < 18 then consumi = 'medio';  
if kpl < 11 then consumi = 'elevato';
```

kpl=15,
consumi=?

kpl>=18 No
11<= kpl < 18 Si

COMANDI IF...THEN...ELSE (2)

kpl=15,
consumi=?

```
if 18 >= kpl      then consumi = 'buono  ' ;  
if 11 <= kpl < 18 then consumi = 'medio  ' ;  
if      kpl < 11  then consumi = 'elevato' ;
```

```
if kpl < 11      then consumi = 'elevato' ;  
else if kpl < 18 then consumi = 'medio  ' ;  
else consumi='buono' ;
```

.....→
Kpl<11? No
Kpl<18? Si

ESEMPIO IF...THEN...ELSE

Si parta dal file canile. csv che si trova nella cartella
v:\carla\sas\

Si costruisca la variabile **classe_eta** aggregando in questo modo:
se ETA<=2 allora **classe_eta=cucciolo**, altrimenti **classe_eta=adulto**.
Controllare se ci sono missing nella variabile ETA: in caso
affermativo porre classe_eta=mancante.

```
DATA canile;  
set canile;  
if eta= . then classe_eta='mancante';  
else if 0 < eta <= 2 then classe_eta='cucciolo';  
else if eta > 2 then classe_eta='adulto';  
proc print;  
run;
```

Trattamento di più variabili

ARRAY var(n|*) lista;

var identifica gruppo variabili
n n. elementi array
lista elenco variabili da raggruppare

Esempio ricodifica missing

```
ARRAY X(10) y1-y10;  
DO i=1 TO dim(x);  
    IF X(i)=99 THEN X(i)=.;  
END;
```

ARRAY

- Le variabili raggruppate devono essere **tutte** dello **stesso tipo** (numeriche o \$);
- La definizione di array non richiede memoria aggiuntiva;
- La variabile array non è memorizzata nel data set

Esempi definizione array

```
ARRAY tot(5) eta sesso titstud x1 x2;  
*ordine nel data set;  
ARRAY tot(*) eta--titstud;  
*variabili carattere;  
ARRAY N(10) $nome1-nome10;
```

Osservazioni con numero variabile di record

- Per SAS ogni record del file corrisponde ad una osservazione;
- Le informazioni relative ad una u.s. possono essere poste su più record (p.e.: un record per ogni esame svolto da studenti).
- Come creare un archivio SAS rettangolare (un record=una u.s.)?

array1.sas

Trasformazione data set: da molti a un record per ogni obs

- Come trasformare un data set sas che contiene **più record** per ogni studente in un data set che contiene **un solo record** per studente?

array2.sas

CICLI

- DO iterativo
- DO WHILE
- DO UNTIL

- DO iterativo

```
DO var=a TO b [BY c];  
...;  
END;
```

var controlla esecuzione ciclo

DO ITERATIVO

```
DO var = a TO b [BY c];  
...  
END;
```

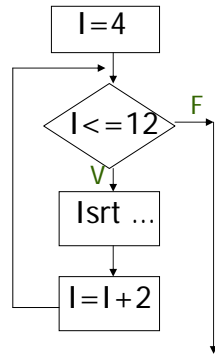
var CONTROLLA ESECUZIONE CICLO
a VALORE INIZIALE var
b " FINALE var
c INCREMENTO di var
 (DEFAULT 1)
CICLO TERMINA QUANDO var > b

DO ITERATIVO: esempio

```
DO I=4 TO 12 BY 2;
...;
END;
```

Iter	I
1	4
2	6
3	8
4	10
5	12

----- uscita
14



DO INTERATIVO con elenco valori

DO var=a,b,c, ...;

La variabile che controlla il ciclo può essere definita da un elenco di valori (numerici o carattere) separati da virgole

```
DO I= 2, 5, 3, 7, 15;
--
END;
```

ITER	1	2	3	4	5
I	2	5	3	7	15

```
DO MESE = 'GEN', 'FEB', 'GIU';
:
END;
```

ITER	1	2	3
MESE	'GEN'	'FEB'	'GIU'

GENERAZIONE DEI DATI DA PROGRAMMA

- Utilizzare DO ITERATIVO;
- Utilizzare OUTPUT per aggiungere una osservazione al data set ad ogni iterazione del ciclo.

Esempi:

doiter1.sas

doiter2.sas

Tabella di contingenza

Tab. 1 – Tabella a doppia entrata

Sesso	Titolo di studio					Totale
	1	2	3	4	5	
Maschi	11	28	11	1	5	56
Femmine	12	30	13	14	13	82
Totale	23	58	24	15	18	138

Scrivere un programma SAS per:

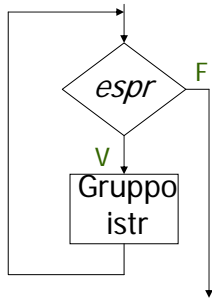
- Input da tabella a lista
- Input da tabella a doppia entrata

doiter4.sas

Tab. 2 – Tabella a lista

Sesso	Titolo di studio	Frequenze
Maschi	1	11
Maschi	2	28
Maschi	3	11
Maschi	4	1
Maschi	5	5
Femmine	1	12
Femmine	2	30
Femmine	3	13
Femmine	4	14
Femmine	5	13

DO WHILE

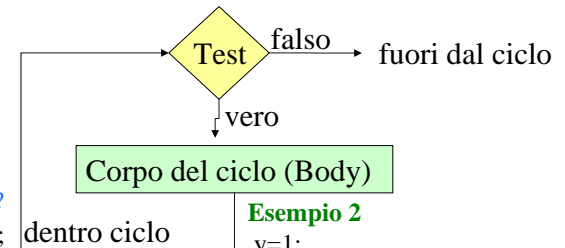


```
DO WHILE (espr);
...
...
END;
```

- n. Iterazioni del ciclo non noto a priori;
- Se espr falsa a inizio ciclo, il ciclo non viene MAI eseguito!

Esempio DO WHILE

```
DO WHILE( espressione);
DO var=start TO stop <BY incremento> WHILE(espr);
```



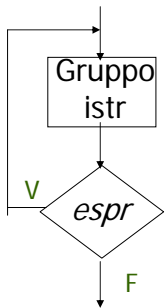
Esempio 1

```
cosa succede
omettendo x=1?
DO WHILE(x<10);
dentro ciclo
x+1;
OUTPUT;
END;
```

Esempio 2

```
y=1;
DO x=1 TO 100 BY 2 WHILE(y<200);
y=y*x; output;
END;
```

DO UNTIL

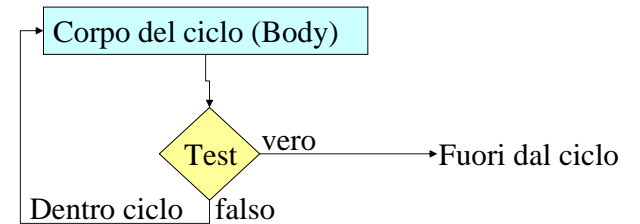


```
DO UNTIL (espr);
...
...
END;
```

Le istruzioni del ciclo vengono eseguite almeno una volta

Esempio DO UNTIL

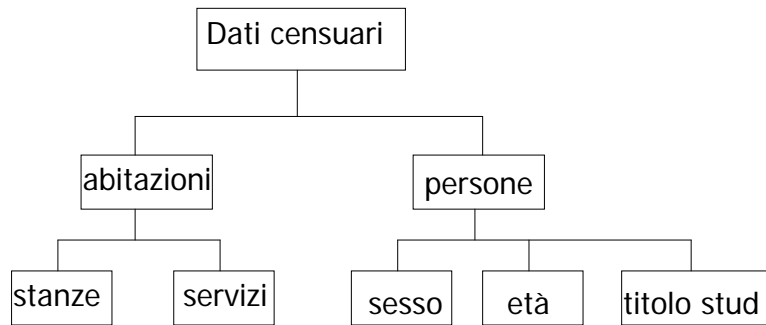
```
DO UNTIL( espressione);
DO var=start TO stop <BY incremento> UNTIL(espressione);
```



```
DATA dati;
x=1;
DO UNTIL (x>10);
OUTPUT;
x+1;
END;
```

cosa succede omettendo o spostando OUTPUT?

Lettura archivi gerarchici



File gerarchico: caso speciale di file con record misti

censimento2.sas

Lettura file con record a lunghezza variabile

- File originale contiene un record per ogni studente;
- Il numero di esami sostenuto da ogni studente è registrato nel campo NESAs;
- Vogliamo leggere voto e codice esame per tutti gli esami sostenuti da ogni studente e creare un record per ogni esame.

input7.sas

Controllo scrittura delle osservazioni nel SAS data set

OUTPUT consente di scrivere le osservazioni in un momento qualsiasi del data step;

DELETE consente di eliminare osservazioni

output1.sas-output3.sas

Provate voi ...

1. Scrivere un programma SAS che generi 20 osservazioni relative ad una variabile X con valori: 0, 2, 4, ..., 20.
2. Definire un ciclo in modo che inizializzata $x=1$ e $y=1$ il ciclo prosegua finché $y<50$ e $x<50$. Ad ogni iterazione del ciclo: X deve essere incrementata di 1 e il valore di Y deve variare nel modo seguente $y=y+x$. Il nuovo data set deve contenere tante osservazioni quante sono le iterazioni del ciclo.

ESERCIZIO 3

creare un SAS dataset dal file `pressione.csv` che si trova nella cartella `v:\carla\sas\`

- Creare le seguenti **variabili**:
 - **Farmaci** presi (considerando come modalità le coppie No No, No Si, Si No, Si Si delle variabili Farmaco_1 e Farmaco_2)
 - **Effetto** sulla pressione (Solo per chi ha preso almeno un farmaco) considerando come modalità le possibili coppie della pressione prima e dopo il farmaco.
- Assegnare un titolo al programma.
- Eliminare tutte le variabili, fuorché quelle appena create e l'identificativo della clinica di appartenenza.

ESERCIZIO 4

Partendo dalla seguente tabella 'vettorializzata':

ID	SESSO	ETNIA	PRESSIONE MAX	PRESSIONE MIN
1	M	bianco	130	60
2	F	nero	140	70
3	F	nero	120	64
4	M	asiatico	150	76
5	M	bianco	124	72
6	M	asiatico	127	80

- ✓ Creare un dataset e salvarlo nella propria libreria;
- ✓ Selezionare gli individui maschi;
- ✓ Creare una nuova variabile come differenza tra la pressione massima e quella minima;
- ✓ Se la differenza è maggiore di 60 porre `Differenza='alta'`, altrimenti `Differenza='bassa'`;
- ✓ Stampare nella finestra OUTPUT i risultati.