الطريقة الثانية :

اذا المتداخلة nested if

```
If logical expression1
Statement 1
Else if logical expression 2
Statement 2
Else if expression 3
Statement 3
Else
Statement 4
end
end
End
```

* بعدد مرات التكرار يتم تكرارها

اذا كان الشرط الأول صحيح فأنه سيقوم بتنفيذ الجملة الأولى , و اذا كان الشرط الأول خطأ فأنه سينفذ الشرط الثاني واذا كان الشرط الثاني خطا فانه سينفذ الشرط الثاالث و اذا كانت الجملة الثالثة خطا فانه سينفذ الجملة التي بعد else و هكذا.

مثال2: اكتب برنامج للمقارنة بين قيم a و b

```
clear
clc
a=input('a=');
b=input('b=');
  if (a>b)
disp ('The result positive value')
  else if (a<b)
disp('The result negative value')
    else
        disp('a=b')
end
end
```

مثال3 : اكتب برنامج يقوم بفحص قيمة المتغيران a , b فاذا كانا اكبر من صفر اوجد حاصل ضربهما وبعكسه اطبع ان القيم ليست اكبر من صفر ؟

الحل :

```
>> a=input('a=') ; b=input(' b=');
 if (a>0 & b>0)
disp (' a and b values greater than zero ');
C=a * b
```

```
else
disp ( ' a and b values not greater than zero ');
end
```

س\\H.W اكتب برنامج يعمل على ايجاد حل المعادلة: $ax^2+bx+c=0$ ؟

تُحل المعادلة بطريقة الدستور

2- ايعاز التكرار for :

لغرض تنفيذ جملة بلغة ماتلاب اكثر من مرة نستخدم ايعازالتكرار for والصيغة العامة هي :

```
For variable =initial value :increment :final value
Statements
End
```

Variable : متغير بلغة ماتلاب
Initial value :القيمة الأولية التي سوف يبدأ بها المتغير
Increment :مقدار الزيادة التي يتم اضافتها الى القيمة الأولية
Final value : مقدارالقيمة النهائية التي يصل اليها المتغير

تعمل جملة التكرار على اخذ القيم الأولية ووضعها في المتغير ثم ينفذ الجمل التي تلي ايعاز التكرار ولحين الوصول الى جملة الأنهاء end
يقوم بأضافة مقدار الزيادة الى المتغير ليحص على قيمة جديدة ويتحقق ان كانت قيمة المتغير هي اصغر من اويساوي القيمة النهائية فاذا كان الجواب نعم يذهب لينفذ من الأعلى كل الجمل التي تلي ايعاز التكرار اما اذا كان الجواب كلا فانه يخرج من التكرار.

مثال1:

```
>>for i=1:5
disp ( i )
end
```

عند التنفيذ سيكون الناتج كما يلي:

```
  i =
1
2
3
4
```

5

مثال2 : اطبع الأعداد التي تقبل القسمة على 3 للقيم من 3 و لغاية 16 .

```
>> for i= 3:3 :16
Disp( i )
End
```

عند التنفيذ سيكون الناتج كما يلي:

```
  i =
3
6
9
12
15
```

مثال3 : اطبع الأعداد تنازلياً ممن 20 ولغاية 17 ؟

```
>> for i=20 : -1 : 17
Disp( i )
End
```

عند التنفيذ سيكون الناتج كما يلي:

```
   i=
20
19
18
17
```

مثال4 : اطبع الأعداد التي تقبل القسمة على 3 (من 16 ولغاية 3) ؟

```
>> clear
clc
for i = 16 : -1 : 3
if rem (i , 3)==0
disp( i)
end
end
```

```
    i =
15
12
9
6
3
```

مثال5 : اوجد مجموع الأعداد من 1 و لغاية 5 ؟

```
>> clear
clc
s=0;
 for i=1 : 5
s=s+i ;
end
disp( s )

Ans =
15
```

مثال6 : اجمع الأعداد الزوجية الواقعة بين 2 و 15؟

```
>> clear
clc
s=0;
 for i=2 : 2: 15
s=s+i ;
end
disp( s )

Ans =
56
```

التكرار المتداخل nested loop

يتكون التكرار المتداخل من اكثر من تكرار و يجب ان يكون غير متقاطع , اما عمله فانه يأخذ اول قيمة من المتغير الأول التابع الى التكرار الأول وينفذ التكرار الثاني ابتداءً من القمة الأولية ولحين الوصول الى اخر قيمة من التكرار الثاني ثم ياخذ القيمة الثانية من التكرار الأول و ينفذ التكرار الثاني ابتداءً من القيمة الأولية الى اخر قيمة وهكذا حتى ينفذ اخر قيمة من التكرار الأول مع التكرار الثاني وصولاً الى اخر قيمة من التكرار الثاني والصيغة العامة هي:

```
For  i =var 1: increment : final value 1
For  j= var 2 : increment :final value 2
Statements
End
End
```

مثال 7:اكتب برنامج يعمل على تكوين مصفوفة ويطلب ابعاد المصفوفة قبل تكوينها :

```
>> n=input( ' n=') ; m= input( 'm=');
for i=1:n
```

```
for j=1:m
        A(i,j)=input('a=');
end
end
```

عند التنفيذ سيطلب قيم n,m التي تمثل ابعاد المصفوفة وقيم a التي تمثل عناصر المصفوفة. فمثلا عند وضع n=2, m=2 سوف يطلب منا ادخال اربع قيم ل a فلو وضعنا قيم a كما يلي :

```
a=1
a=2
a=3
a=4
```

سيصبح شكل المصفوفة كالتالي:

```
A=
1     2
3     4
```

س \\ اكتب برنامج لأيجاد جدول الضرب لأي قيمة معطاة ؟

3- ايعاز التنفيذ المشروط while

لغرض تكرار جملة معينة اكثر من مرة نستخدم ايعاز while ويعمل هذا الأيعاز عمل التكرار for والصيغة هي :

```
While condition
Statements
Increment
End
```

Condition : اذا كان الشرط صحيح فأنه ينفذ كل الجمل التي تلي while وبعكسه فأنه ينتقل لينفذ الجمل التي بعد end .
Increment :مقدار الزيادة التي تضاف على العداد لكي يستمر في التنفيذ لحين الوصول الى اخر قيمة
هذه الأداة يمكن معرفة أو تلمس عملها من معناها اللغوي (بما ان) حيث ان عملها يشبه الى حد ما عمل التكرار , لكن بدون مقدار للزيادة أو النقصان للوصول الى نقطة النهاية , فقط يكون الشرط هو علامة أو نقطة الأنتهاء للدالة.

While هنا يوضع الشرط المتغير ما ينتهي عمله بشرط ما , وعادةً ما يمثل الشرط بالعمليات المنطقية

مثال1 : اطبع الأرقام من 1 ولغاية 5 ؟

```
>>  s=1;
while ( s <=  5 )
disp( s);
s=s+1;
end
```

```
--------------------------
1
2
3
4
5
```

قبل الدخول بأيعاز while يجب ان نعرف القيمة الأولية الى العداد s وفي هذا المثال عرفت الأولية بالقيمة 1 , وبما ان القيمة الأولية اصغر من القيمة النهائية فان الشرط صحيح وسوف ينفذ الجمل التي تلي ايعاز while ,يستمر لحين الوصول الى القيمة النهائية

مثال : اطبع الأعداد الزوجية بين 2 و8 ؟

```
>> s= 2;
while s<=8
disp( s )
s=s+2;
end
```

```
2
4
6
8
```