LECTURE (9)

عبارة (continue & break) :

تستخدم عبارة continue لإهمال الحالة الحالية من الحلقة والدارة , أما عبارة break فتستخدم لقطع (الدارة) والخروج منها مباشرة وتوجه التنفيذ الى الجملة التي تلي كلمة (end) التي تعود للحلقة المقطوعة.

ملاحظة//

1. وظيفة break هي إنهاء تنفيذ حلقة for أو while والخروج من الحلقة.
2. التعبير continue يسمح لنا بتخطي جزء من حلقة التكرار عند حدوث عامل خارجي، لكن إكمال بقية الحلقة إلى نهايتها .بعبارةٍ أخرى: سينتقل تنفيذ البرنامج إلى أوّل حلقة التكرار عند تنفيذ التعبير. continue

Example //: Write a program in MATLAB to print integers from 30-15 excluding(مستثنياً) the values 20,25 and 30.

Solution: M-file

```
clear;
clc;
close all;
for i = 30: -1:15
 if (i== 20) | (i == 25) | (i == 30)
 continue
```

```
 else
i  %disp(i)
 end
 end
```

Example // Write a program in MATLAB to print numbers that are not divisible by (2) from 20-50 (meaning that only the odd numbers are shown to us)

```
 clear;
 clc;
 close all;
  for x=20:1:50      % for x=21:2:50
   if rem(x,2)==0;
    continue
   end
 pause (1)
 %disp('x')
 fprintf('v_value  : %d/n',x)
 end
```

مكرم الجبالي

Example //: Write a program in MATLAB to read 50 numbers, stop reading if any Zero number appears, and print that number.

```
clear;
 clc;
close all;
 for m = 1: 50
 x = input ('x=')
 if x == 0
 disp(x)
 break
 end
 end
```

Example // A program in MATLAB that prints numbers divisible by 7 from 1 to 50:

```
clear;
 clc;
close all;
  for n = 1:50
```

كريم الجبالي

```
    if mod(n,7)
      continue
    end
    disp('Divisible by 7: ')
end
```

- **mod (modulus):**
  - يُرجع باقي القسمة **إيجابيًا دائمًا** عند القسمة على عدد موجب.
  - يُحسب باقي القسمة بناءً على **تقريب الناتج نحو الصفر**.
- **rem (remainder):**
  - يُرجع باقي القسمة بناءً على **إشارة العدد الأول (المقسوم)**.
  - يُحسب باقي القسمة بناءً على **تقريب الناتج نحو الصفر أيضًا**.

مقارنة رياضية:

عندما تكون الأرقام موجبة:

- **mod** و **rem** يعطيان نفس النتيجة.

```
mod(10, 3)  % الناتج: 1
rem(10, 3)  % الناتج: 1
```

عندما تكون الأرقام سالبة:

- النتائج تختلف بناءً على الدالة.

```
mod(-10, 3)  % الناتج: 2
rem(-10, 3)  % الناتج: -1
```

عندما يكون القاسم سالبًا:

- النتيجة تعتمد على الدالة:

```
mod(10, -3)  % الناتج: -2
rem(10, -3)  % الناتج: 1
```