

برمجة إجرائية

جلسة عملي

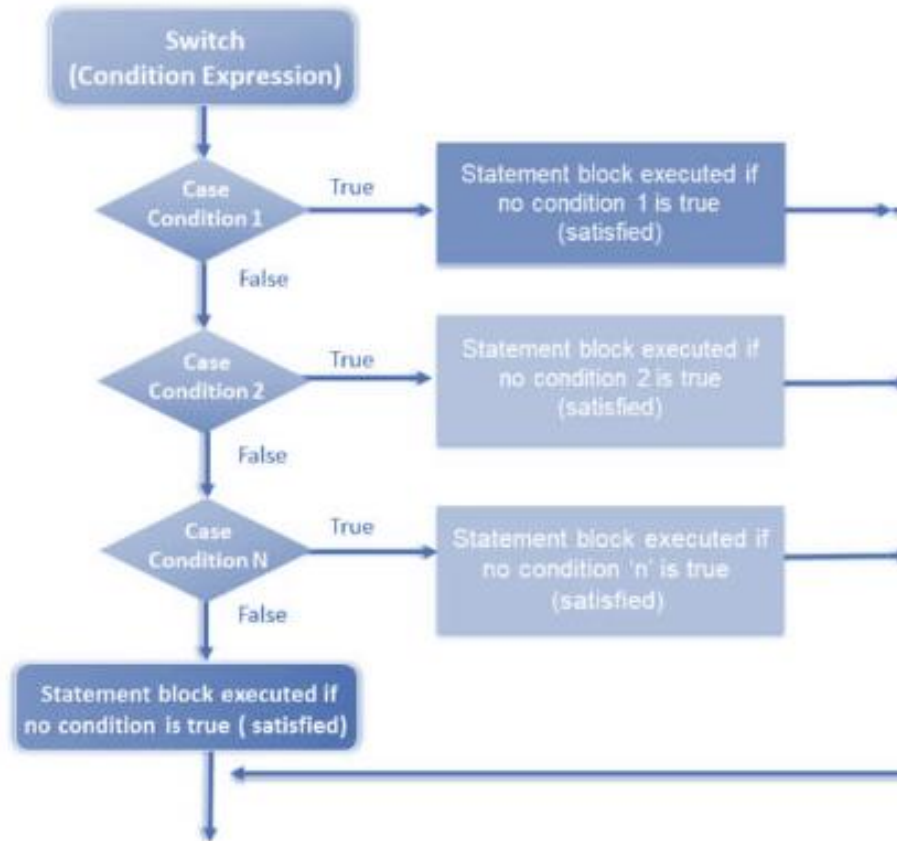
While - Switch

إشراف:

د. عيسى الغنام

إعداد:

م. همam ياسين م. آية خيربك



```
switch switch_expression  
  case case_expression  
    statements  
  case case_expression  
    statements  
  ...  
  otherwise  
    statements  
end
```

```
n = input('Enter a number: ');

switch n
    case -1
        disp('negative one')
    case 0
        disp('zero')
    case 1
        disp('positive one')
    otherwise
        disp('other value')
end
```

تمرين:

اكتب برنامج يقوم فيه المستخدم بإدخال A,B,C,D,F وهي تعبر عن المعدل، فيقوم البرنامج بحساب تقدير هذا المعدل كالتالي:

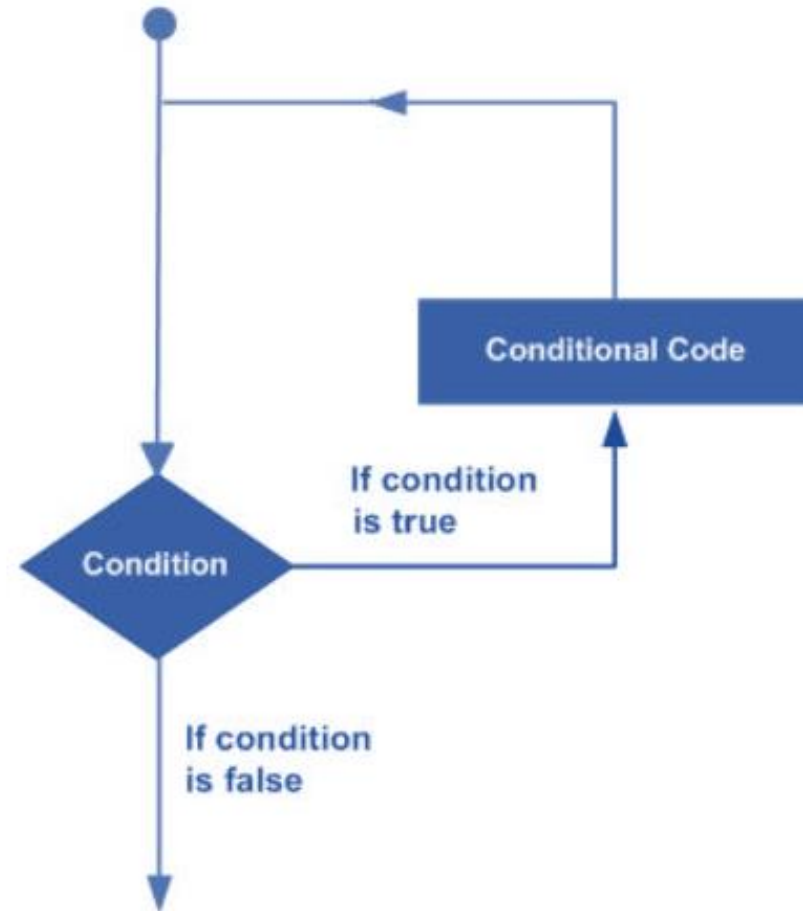
A	B	C	D	F	Otherwise
Excellent!	Very Good	Well Done	You Passed	Better Try Again	Invalid Grade

تمرين:

اكتب برنامج يقوم فيه المستخدم بإدخال A,B,C,D,F وهي تعبر عن المعدل، فيقوم البرنامج بحساب تقدير هذا المعدل كالتالي:

A	B	C	D	F	Otherwise
Excellent!	Very Good	Well Done	You Passed	Better Try Again	Invalid Grade

```
grade = input('enter grade:', 's');
switch grade
case 'A'
| fprintf('Excellent!\n' );
case 'B'
| fprintf('Very Good \n' );
case 'C'
| fprintf('Well done\n' );
case 'D'
| fprintf('You passed\n' );
case 'F'
| fprintf('Better try again\n' );
otherwise
| fprintf('Invalid grade\n' );
end
```



```
while <expression>  
    <statements>  
end
```


ما خرج هذا البرنامج:

```
a = 10;  
% while loop execution  
while( a < 20 )  
    fprintf('value of a: %d\n', a);  
    a = a + 1;  
end
```

تمرين:

اكتب برنامج يقوم فيه المستخدم بإدخال عدد صحيح، فيقوم البرنامج بحساب $n!$ لهذا الرقم وإظهار الناتج على الخرج.

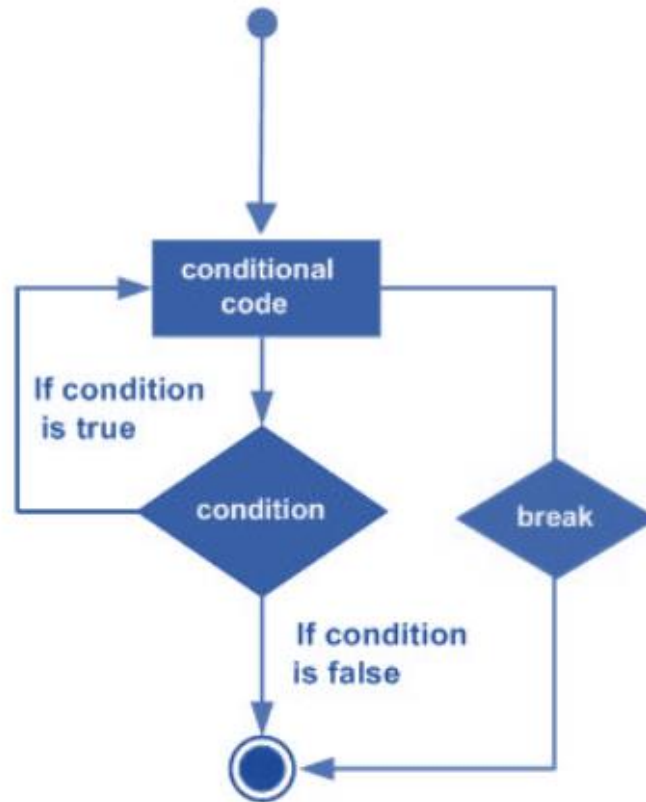
تمرين:

اكتب برنامج يقوم فيه المستخدم بإدخال عدد صحيح، فيقوم البرنامج بحساب $n!$ لهذا الرقم وإظهار الناتج على الخرج.

```
n = 10;  
f = n;  
while n > 1  
    n = n-1;  
    f = f*n;  
end  
disp(['n! = ' num2str(f)])
```

تعليلة break:

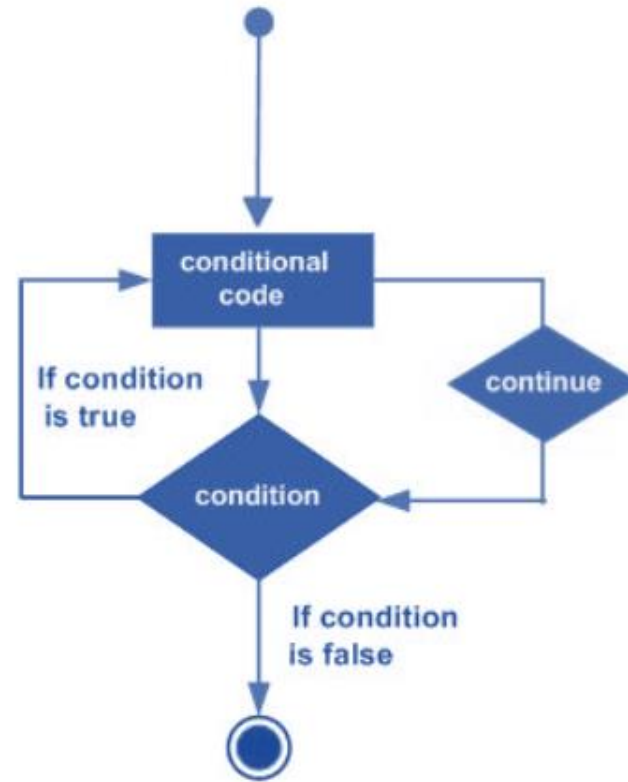
تقوم هذه التعليلة بإنهاء تنفيذ الحلقة أو الشرط ومتابعة العمل بعدها.



```
y = [-2 -4 0 -4 3 7];  
for i = 1 : length(y)  
    if y(i) > 0  
        break  
    end  
    z = y(i) + 6;  
    disp(z)  
end
```

تعليلة :continue

تقوم هذه التعليلة بالانتقال إلى التنفيذ التالي للحلقة.



```
for i = 1 : 7  
    if (i == 2 | i == 5)  
        continue  
    end  
    disp(i)  
end
```

تعليمة break ,continue :

ماذا سيكون الخرج النهائي لهذا البرنامج؟

```
clc  
clear  
n=0;  
for x=1:55  
    n=n+1;  
    continue  
    break;  
end  
disp(n)
```


تمرين:

اكتب برنامج يقوم بتوليد مصفوفة من أعداد صحيحة بين الـ 0 و 50 بقياس 5x5 ثم يقوم البرنامج بالبحث عن رقم 23 في المصفوفة ثم يقوم بإرجاع عنوان هذا العدد في المصفوفة.

تمرين:

```
arr = randi([0 50],5);  
index=[-1 -1];  
for i=1:5  
    for j=1:5  
        if(arr(i,j) == 23)  
            index = [i j];  
            break  
        end  
    end  
    if(index(1)~= -1)  
        break  
    end  
end  
x=input('x');  
disp(arr)  
disp("index is = ")  
disp(index)
```

تمرين:

اكتب برنامج يقوم فيه المستخدم بإدخال عدد صحيح، فيقوم البرنامج بحساب أصغر قاسم لهذا العدد ويظهره.

```
3 num = input('num=');
4 i=2;
5
6 %starting the while loop
7 while true
8     rem = mod(num,i);
9     |
10    % Condition for break
11    if(rem==0)
12        break
13    end
14    %incrementing i
15    i=i+1;
16 end
17
18 % Displaying the divisor
19 disp(i)
```

تمرين:

اكتب برنامج يقوم فيه المستخدم بإدخال عناصر مصفوفة ثنائية A بقياس 3×3 ثم نقوم بالمرور على عناصر المصفوفة كل عنصر على حدى ونقوم باختبار هذا العنصر إذا كان فردياً نضع مكانه 1 وإذا كان زوجياً نضع مكانه 0 ونخزن المصفوفة الجديدة باسم B.

تمرين:

اكتب برنامج يقوم فيه المستخدم بإدخال عناصر مصفوفة ثنائية A بقياس 3x3 ثم نقوم بالمرور على عناصر المصفوفة كل عنصر على حدى ونقوم باختبار هذا العنصر إذا كان فردياً نضع مكانه 1 وإذا كان زوجياً نضع مكانه 0 ونخزن المصفوفة الجديدة باسم B.

```
A=randi(3,3);
B=A;

for i=1:3
    for j=1:3
        A(i,j)=input('enter num');
    end
end

for i=1:3
    for j=1:3
        if mod(A(i,j),2)==0
            B(i,j)=0;
        else
            B(i,j)=1;
        end
    end
end

disp(A);
disp(B);
```