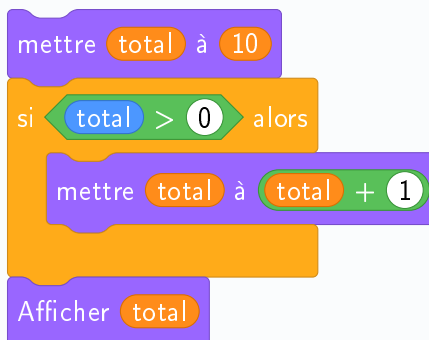
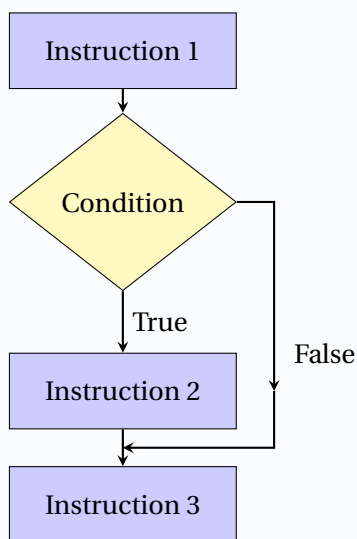


AP03

CONDITIONS IF

Les structures conditionnelles IF



```
# En python
Instruction_1
if condition:
    Instruction_2
Instruction_3
```

```
# En python
total = 10
if total > 0:
    total = total + 1
print(total)
```

- l'Instruction 1 est effectuée
- SI la Condition est True alors l'Instruction 2 est effectuée
- SINON on va directement à l'Instruction 3
- Dans tous les cas l'Instruction 3 est effectuée

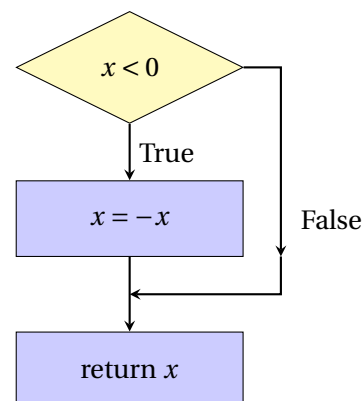
La Condition est un test qui est évalué en python par un booléen.

Exemple : La condition IF sans ELSE : la valeur absolue d'un nombre

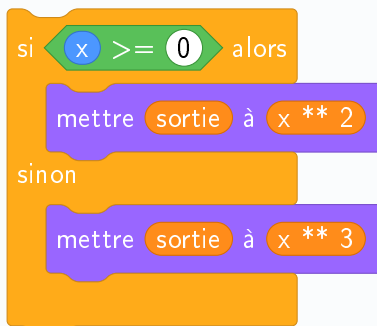
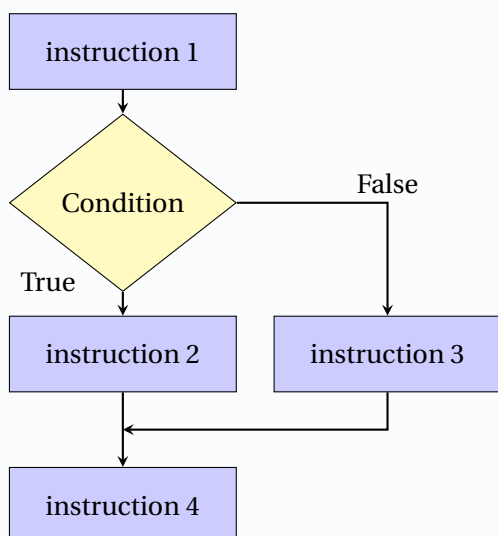
- si $x \geq 0$, alors $|x| = x$
- si $x < 0$, alors $|x| = -x$

</> Code Python

```
1 def valeur_absolue(x):
2     ''' Renvoie |x| '''
3     if x < 0:
4         # si x < 0 on échange x par -x
5         x = - x
6     return x
```



Les structures conditionnelles IF – ELSE



```
# En python
instruction_1
if condition:
    instruction_2
else:
    instruction_3
instruction_4
```

- l'instruction 1 est effectuée
- SI la Condition est True alors l'instruction 2 est effectuée
- SINON l'instruction 3 est effectuée
- Dans tous les cas l'instruction 4 est effectuée

Exemple : La condition IF – ELSE

</> Code Python

```

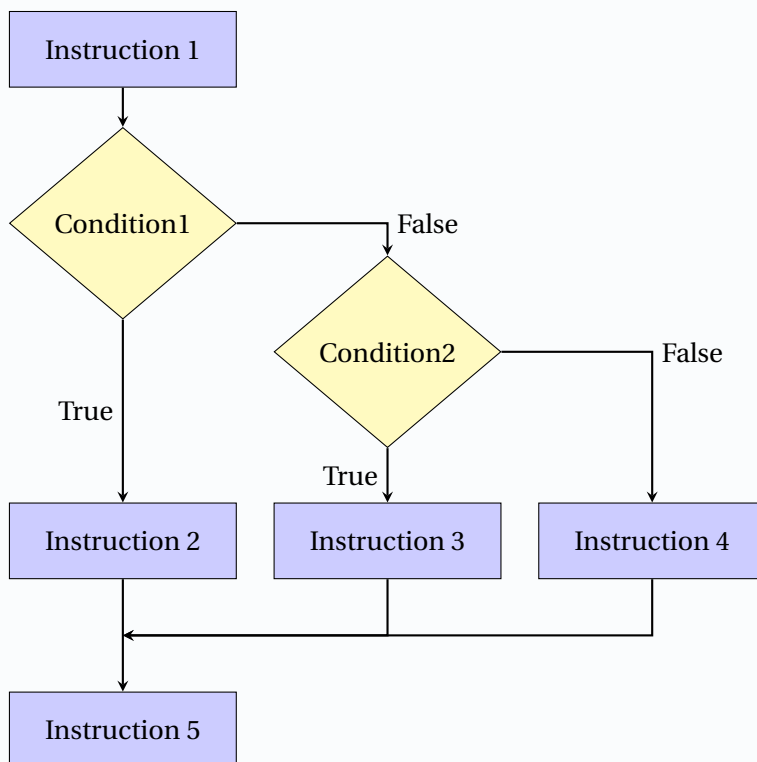
1 def ma_fonction(x):
2     if x >= 0:
3         sortie = x ** 2
4     else:
5         sortie = x ** 3
6     return sortie

```

**Les structures conditionnelles IF – ELIF – ELSE**

Dans le cas où il y a plusieurs possibilités de réponse, on peut utiliser deux structures différentes en python. On suppose une fonction prenant comme paramètre une variable x de type entier.

$$f(x) = 1 \text{ si } x > 0 \quad f(x) = 0 \text{ si } x = 0 \quad f(x) = -1 \text{ si } x < 0$$



</> Code Python

```

1 def ma_fonction(x):
2     if x > 0:
3         sortie = 1
4     elif x == 0:
5         sortie = 0
6     else:
7         sortie = -1
8     return sortie

```

</> Code Python

```

1 def ma_fonction(x):
2     if x > 0:
3         sortie = 1
4     else:
5         if x == 0:
6             sortie = 0
7         else:
8             sortie = -1
9     return sortie

```

Exercice 1 : Compléter les fonctions suivantes

</> Code Python

```

1 def plus_grand2(a, b):
2     ''' Renvoie le plus grand des
3         deux nombres a et b '''
4
5
6
7
8
9
10
11
12 assert plus_grand2(3, 5) == 5
13 assert plus_grand2(3, -5) == 3

```



</> Code Python

```

1 def plus_grand3(a, b, c):
2     ''' Renvoie le plus grand des
3         trois nombres a, b et c '''
4
5
6
7
8
9
10
11
12 assert plus_grand3(3, 5, 2) == 5
13 assert plus_grand3(3, -5, 2) == 3

```

