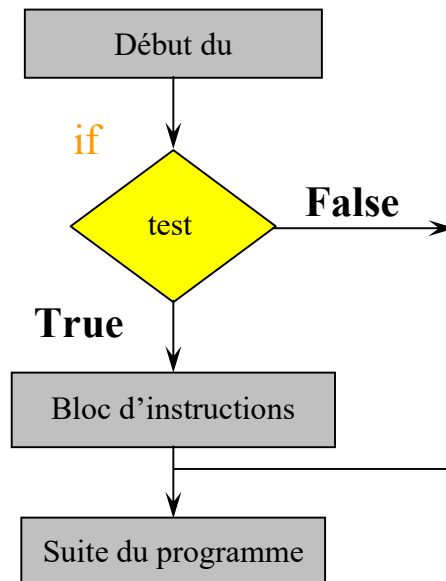


CHAPITRE 2 : LES CONDITIONS

I. L'instruction if.



Syntaxe :

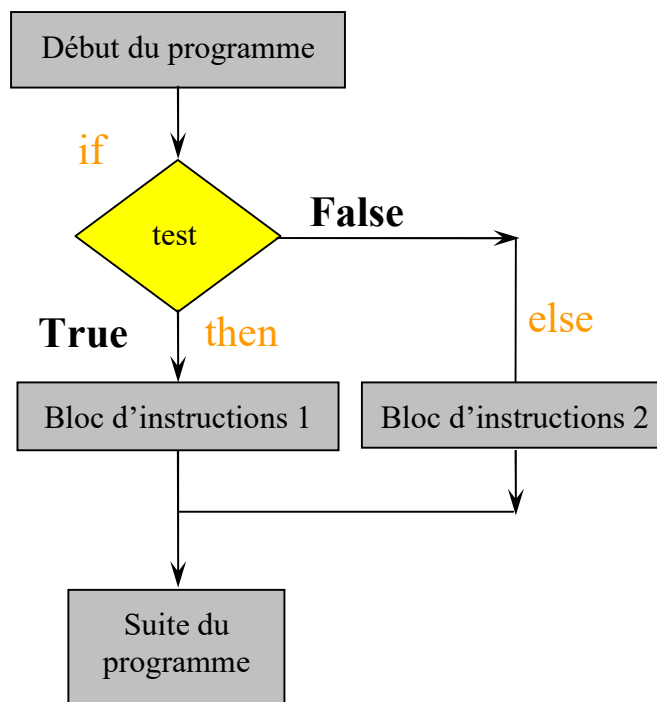
```

if test:           # ne pas oublier le signe de ponctuation '!'
    bloc d'instructions  # attention à l'indentation
# suite du programme
    
```

Si le test est vraie (True) alors le bloc d'instructions est exécuté.

Si le test est fausse (False) on passe directement à la suite du programme.

II. L'instruction else.



Syntaxe :

```

if test:
    
```

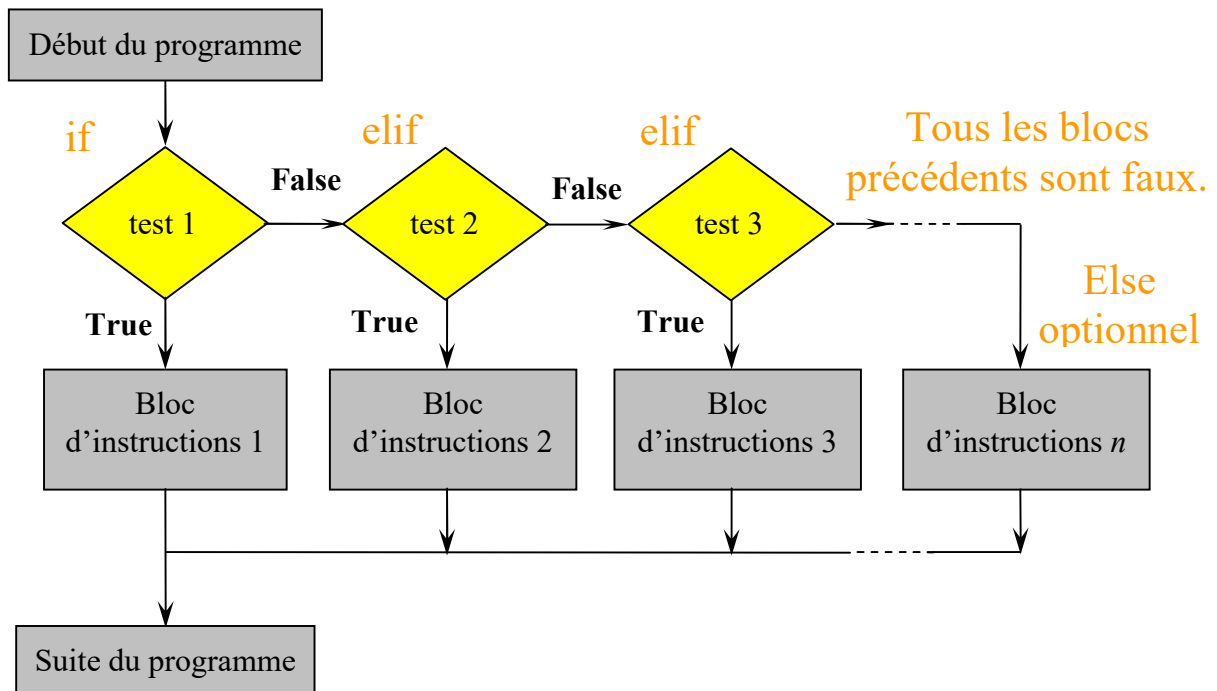
```

    bloc d'instructions 1
else:
    bloc d'instructions 2
# suite du programme
    
```

attention à l'indentation
else est au même niveau que if
attention à l'indentation

Une instruction else est toujours associée à une instruction if.
Si le test est vraie (True) alors le bloc d'instructions 1 est exécuté.
Si le test est fausse (False) alors c'est le bloc d'instructions 2 qui est exécuté.

III. L'instruction elif.



Syntaxe :

```

if test 1:
    bloc d'instructions 1
elif test 2:
    bloc d'instructions 2
elif test 3:
    bloc d'instructions 3
else:
    bloc d'instructions 4
# suite du programme
    
```

ici deux instructions elif, mais il n'y a pas de limitation

Si le test 1 est vrai alors le bloc d'instructions 1 est exécuté, et on passe à la suite du programme.

Si le test 1 est faux alors on évalue le test 2 :

- si le test 2 est vraie on exécute le bloc d'instructions 2, et on passe à la suite du programme.
- si le test 2 est faux alors on évalue le test 3, etc...

Le bloc d'instructions 4 est donc exécuté si tous les tests sont faux (c'est le bloc "par défaut").

Une instruction elif (contraction de else if) est toujours associée à une instruction if.

Parfois il n'y a rien à faire, dans ce cas, on peut omettre l'instruction else :

L'instruction elif évite souvent l'utilisation de conditions imbriquées (et souvent compliquées).