

Nous réutiliserons dans ce TP les instructions et les notions vues au TP1 auxquelles nous ajouterons quelques tests conditionnels.

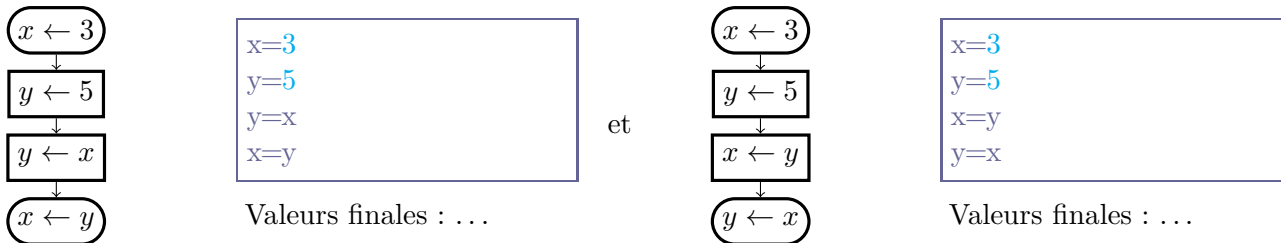
On continue à écrire sous pyzo à l'aide de l'interpréteur de scripts.

## I Séquences d'instructions

Comme en mathématiques, selon l'ordre dans lequel on effectue les actions, le résultat peut être différent.

expl :  $(1 + 2) \times 3 \neq 1 + (2 \times 3)$

Comparer les deux algorithmes suivants :



En effet, il faut voir les variables comme des boîtes contenant une unique valeur. Dessiner schématiquement les actions faites dans le 1<sup>er</sup> algorithme.

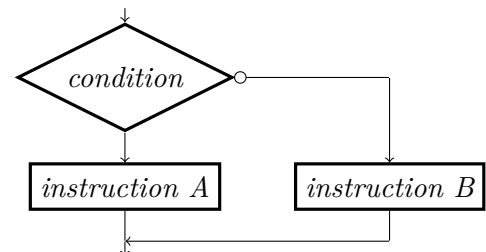
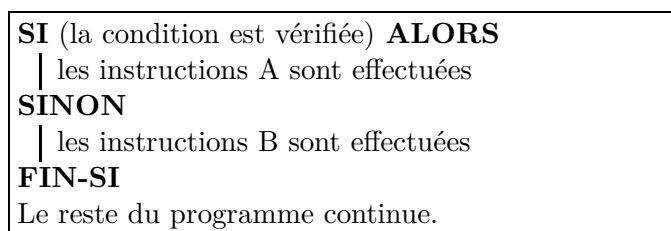
### Exo 1 :

1. Proposer un algorithme qui échange les valeurs de x et de y que l'utilisateur aura préalablement entrées.
2. Implémenter cet algorithme sous python.

## II Tests conditionnels

On souhaite effectuer des tests afin de conditionner nos actions aux résultats de ce ou ces tests.

L'algorithme suivi sera alors le suivant :



Sous python, la structure de test binaire se présente basiquement ainsi

```

if (condition):
    bloc d instructions A
else :
    bloc d instructions B
instructions suivantes
  
```

et si on souhaite faire un test comportant plus de deux possibilités,

```

if (condition 1):           # test conditionnel
    bloc d instructions 1   # est effectue si la condition 1 est vraie
elif (condition 2) :       # 2e test conditionnel
    bloc d instructions 2   # est effectue si la condition 2 est vraie
elif (condition 3) :       # 3e test conditionnel
    bloc d instructions 3   # est effectue si la condition 3 est vraie
else :
    bloc d instructions 4   # est effectue si toutes les conditions ont ete
                            fausses
instructions suivantes      # instructions a l'exterieur du bloc conditionnel
    
```

expl : On souhaite demander à une personne son genre,

```

R=input("Veuillez entrer 'F' si vous etes
        une femme et 'H' si vous etes un homme.")
if (R=='F'):                # test conditionnel
    print("Vous etes une femme")
else :                      # sinon
    print("Vous etes un homme")
print("Bienvenue !")
    
```

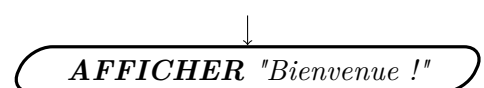
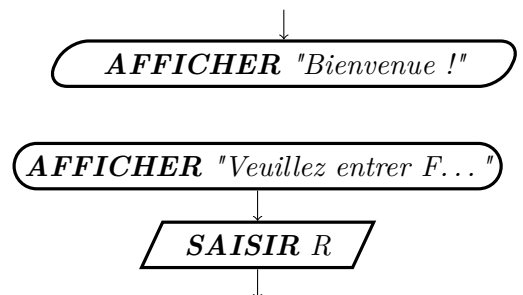
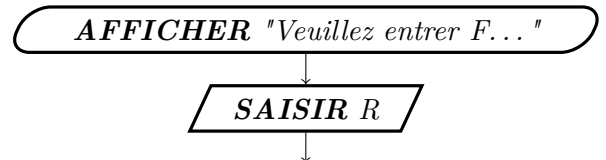
1. préciser le type de la variable R.
2. proposer un organigramme de cet algorithme.
3. quel est le défaut de cet algorithme ?

expl (suite) :

On protège donc la réponse du programme en ajoutant un test. Compléter le programme et l'organigramme.

```

R=input("Veuillez entrer 'F' si vous etes
        une femme et 'H' si vous etes un homme.")
if (R=='F'):                # 1er test conditionnel
    print("Vous etes une femme")
elif (.....) :            # 2e test conditionnel
    print("Vous etes un homme")
else :                      # sinon
    print("Vous n'avez pas correctement
          repondu")
print("Bienvenue !")
    
```



**Exo 2 :**

```

a=float(input("Entrer la longueur du cote adjacent suppose. "))
b=float(input("Entrer la longueur du cote oppose suppose. "))
c=float(input("Entrer la longueur de l'hypothenus suppose. "))

if ( ... == ... ):
    print("Le triangle n'est pas ... comme vous le supposez.")
else :
    print(" ... ")

```

1. À quoi sert le mot `float` avant la fonction `input()` ?
2. proposer un organigramme correspondant à cet algorithme et compléter le code.
3. Implémenter le code suivant afin de vérifier si un triangle est rectangle (*e.g.* 3, 4, 5 et 3, 4, 6).

**III Les booléens**

Nous avons vu que les variables possédaient un type sous python. Entrer le code suivant et exécuter-le.

```

x=1
print("x est du type",type(x))
y="il faut se mefier des petites differences"
print("y est du type",type(y))
z= x==1.
print("z est du type",type(z))

```

Dans un test conditionnel, quel est le type utilisé ?

Confirmez votre intuition en reprenant l'exemple du genre de la personne, et entrer `type(R=='F')`.

**Définition 1**

Un booléen est une entité qui ne peut prendre que deux valeurs possibles, souvent vrai ou faux.

En langage informatique, il est parfois codé par un entier : 0 pour faux,  $\neq 0$  pour vrai. Lorsque le type `boolean` existe, il FAUT utiliser ce type précis (ce qui est le cas en python).

Entrer ces instructions dans le shell et étudier les réponses.

entrée

réponses

```

x=3
x==3
x!=5
x>=2
"1" in "123"
x==0
x<5
x==0 and x<5
x==0 or x<5

```

**Exo 3 :**

On propose un début de code qui demande un mot de passe. On souhaite

1. si le mot de passe entré est 'secret', le mot secret est affiché ;
2. si le mot de passe est une partie du mot "secret", un message annonce qu'il y a eut une erreur de frappe (*e.g.* si "secr" est entré on annonce une erreur de frappe) ;
3. sinon on affiche un message courtois de refus.

Compléter ce programme.

```

mot2passe=input(" Entrer votre mot de passe ")
if (mot2passe == "secret "):
    print(" secret ")
elif ( mot2passe ..... ) :
    print(" ..... ")
else :
    print(" ..... ")
    
```

**Exo 4 :** On donne l’algorithme ci-contre :

1. Quelle valeur est affichée en sortie si on entre :
  - 112 : ...
  - 250 : ...
  - 321 : ...
2. déterminer l’organigramme correspondant
3. Une agence de location, Autoloc’, possède un logiciel permettant de calculer le prix (en euro) de location d’une voiture à partir du nombre de kilomètres parcourus. Le logiciel utilise l’algorithme décrit ci-contre.  
En déduire les informations manquantes dans l’encadré suivant :

<p><b>Variables</b></p> <p><i>km</i> : entier</p> <p><i>prix</i> : réel</p>
<p><b>Algorithme</b></p> <p><b>SAISIR</b> <i>km</i></p> <p><b>SI</b> <math>km &lt; 250</math> <b>ALORS</b></p> <p style="padding-left: 20px;">  <math>prix \leftarrow 99</math></p> <p><b>SINON</b></p> <p style="padding-left: 20px;">  <math>prix \leftarrow 99 + (km - 250) \times 0,34</math></p> <p><b>FIN-SI</b></p> <p><b>AFFICHER</b> <i>prix</i></p> <p><b>Fin Algorithme</b></p>

AUTOLOC ’

Tarif de location : ..... €

Ce tarif permet de parcourir ..... kms

puis ..... € par km supplémentaire.

Certaines de ces voitures sont équipées de la climatisation, l’agence demande alors un supplément de 0,10 € pour chaque kilomètre parcouru, la climatisation augmentant sensiblement la consommation en carburant.

3. modifier l’organigramme puis l’algorithme précédents afin que :
  - On demande si le client souhaite la climatisation.
  - Le montant total de la location soit calculé et affiché.
4. Implémenter l’algorithme en python et le tester sur quelques entrées.

### Travail à faire pour la prochaine séance

**Exo 5 :** Proposer un organigramme puis un algorithme qui, à partir d’une configuration de Thalès avec A, B, C d’une part, A, D, E d’autre part, demande à l’utilisateur d’entrer les longueurs AB, AC, AD et AE et conclut si les droites (BC) et (DE) sont parallèles.

Implémenter alors votre algorithme à l’aide de python.

**Exo 6 :** Proposer un organigramme puis un algorithme qui, étant donné les coefficients a, b, c d’une fonction polynômiale du 2<sup>n</sup> degré  $f(x) = ax^2 + bx + c$ , calcul le discriminant ( $\Delta$ ) et renvoie la valeur de  $\Delta$  accompagné des deux racines, d’une seule ou aucune selon le cas.

Implémenter alors votre algorithme à l’aide de python.