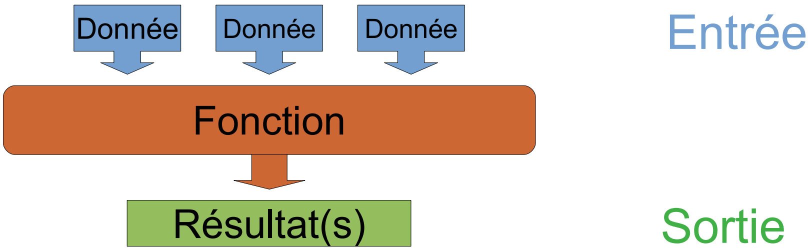
Algorithmique 3 : Les fonctions en Python

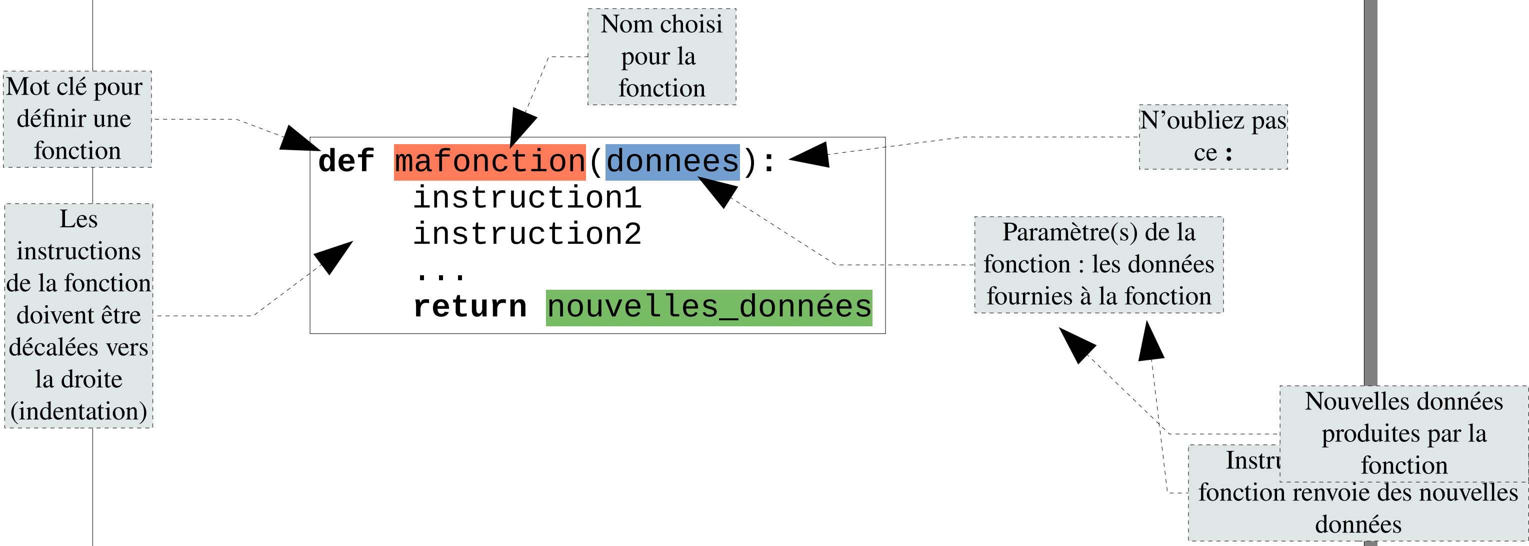
Nous avons déjà utilisée des fonctions prédéfinies dans Python : print et input.

Voyons maintenant comment fabriquer nos propres fonctions.

Une fonction en Python est en fait… un programme !
Elle reçoit donc des données en entrée et produit un résultat (affichage, dessin, nouvelles données) en sortie.



Pour définir une fonction en Python, il faut utiliser le mot clé def.



Un programme bien écrit est découpé en plusieurs fonctions (plusieurs sous-programmes).

Exercice I : définition et appel d'une fonction

1°) Reprenons le programme de l'exercice II de la série Algorithme 2 :

```
y = x + 2
z = y*3
a = z**2
print(a)
```

Ce programme (appelons-le `mafonction`) a besoin d'une valeur pour la variable *x* en entrée et nous voulons récupérer *a* en sortie.

Entrez ce programme dans l'éditeur :

```
def mafonction(x):
     y = x + 2
     z = y*3
     a = z**2
     return a
```

2°) Exécutez votre programme. Rien ne se passe, c'est normal !

Nous avons défini une fonction, il faut maintenant l'appeler (l'exécuter). Pour cela, tapez ceci dans la console Python puis validez avec OK :

```
>>> mafonction(5)
```

La touche permet d'éviter de taper le nom de la fonction

pour demander l'exécution de la fonction « `mafonction` » avec comme valeur 5 pour x.

Vérifiez les valeurs obtenues dans le tableau de l'exercice II (astuce : utilisez la flèche vers le haut).

3°) La valeur renvoyée par la fonction peut être récupérée dans une variable. Essayez ceci :

```
>>> a = mafonction(5)
>>> b = mafonction(7)
>>> a + b
```

Ne confondez pas *définition* et *appel* d'une fonction :

- définir une fonction (avec `def mafonction(…):`) : l'enregistrer dans la mémoire de la calculatrice, dans un fichier de mon ordinateur ;
- appeler une fonction (avec `mafonction(…)`) : l'exécuter en lui fournissant les données dont elle a besoin.

Exercice II : hypoténuse

Nous voulons écrire une fonction donnant la longueur de l'hypoténuse d'un triangle rectangle.

1°) Combien de nombres y a-t-il en entrée de la fonction ? Et en sortie de la fonction ?

2°) Complétez ce programme(*) :

```
def hyp(………………):
     res = ……………………………
     return ……………
```

(*) quelques indications :

- s'il y a plusieurs valeurs en entrée, les séparer par une virgule (qui est sur la 5ième touche de la 2ième ligne) ;
- la racine carrée s'écrit `sqrt()`, par exemple, la racine carrée de 3 s'écrit `sqrt(3)`.

3°) Entrez ce programme dans la calculatrice et testez la fonction avec les valeurs suivantes :

Côtés de l'angle droit	1 et 1	3 et 4	$\sqrt{5}$ et 2	33 et 56	16 et 63
Longueur(**) de l'hypoténuse					

(**) approximative…

Exercice III : distance entre deux points

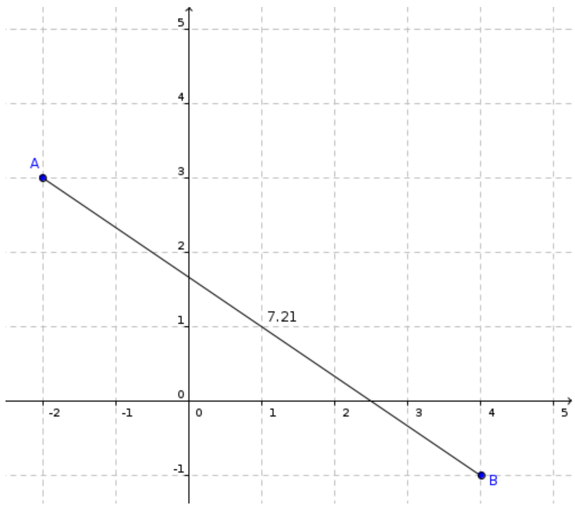
Pour calculer la distance entre deux points A et B (dans un repère orthonormé) nous avons besoin de leurs coordonnées.

Nous pouvons alors calculer les deux coordonnées du vecteur $\overrightarrow{AB}$ puis la norme de ce vecteur.

1°) Complétez cette fonction (utilisez la fonction `hyp` de l'exercice IV) :

```
def dist(xA, xB, yA, yB):
     X = ........................................
     Y = ........................................
     n = ........................................
     return .................
```

2°) Ajoutez cette fonction dans la calculatrice, dans le script de l'exercice IV, et testez la fonction avec les points suivants :



3°) Ajoutez une fonction qui renvoie les coordonnées du milieu d'un segment puis testez-là avec les deux points ci-dessus (pensez à vérifier votre réponse avec le graphique).

Exercice IV

Soient trois points A, B, C dont les coordonnées sont connues dans un certain repère.

1°) Écrivez un algorithme en pseudo-code qui, à partir des coordonnées de ces trois points, donne les coordonnées du point D tel que $ABCD$ soit un parallélogramme.

2°) Traduisez le en programme que vous entrerez dans votre calculatrice.

Testez-le avec les points suivants : A (– 1 ; 3), B (2 ; – 3), C (4 ; 2).

Exercice V

Son nom est à l'origine du mot « algorithme » !

Ce terme est devenu « algèbre » en français !

Problème d'Al-Khwarizmi tiré de « Kitab al-Mukhtasar fi Hisab al-Jabr wa al-Muqàbala » (vers 825) فأما الأموال والجذور التي تعدل العدد فمثل قولك مال وعشرة أجذاره يعدل تسعة وثلاثين درهما ومعناه أي مال اذا زدت عليه مثل عشرة أجذاره بلغ ذلك كله تسعة وثلاثين . فبابه(١) أن تنصف الأجذار وهي في هذه المسألة خمسة فتضربها في مثلها فتكون خمسة وعشرين فتزيدها على التسعة والثلاثين فتكون أربعة وستين فتأخذ جذرها وهو ثمانية فتنقص منه نصف الأجذار هو خمسة فيبقى ثلاثة وهو جذر المال الذي تريد والمال تسعة .	« Un bien et dix de ses racines égalent trente-neuf dirhams ». En termes modernes : trouver le nombre x tel que $x^2 + 10x = 39$.

« Son procédé de résolution consiste à diviser les racines par deux, et c'est cinq dans ce problème.	$10 \div 2 = 5$
Tu le multiplies par lui-même et ce sera vingt-cinq.	$5 \times 5 = 25$
Tu l'ajoutes à trente-neuf. Cela donnera soixante-quatre.	$25 + 39 = 64$
Tu prends alors sa racine carrée qui est huit	$\sqrt{64} = 8$
et tu en retranches la moitié [du nombre] des racines qui est cinq.	$8 - 5 = 3$
Il reste trois et c'est la racine que tu cherches et le carré est neuf. » (source : http://trucsmaths.free.fr/alkhwarizmi.htm)	Donc $x = 3$.

Vérifions cela : $3^2 + 10 \times 3 = 39$ effectivement.

1°) Écrire un algorithme correspondant à la résolution de l'équation $x^2 + px = q$ par la méthode d'Al-Khwarizmi (pensez à bien stocker les résultats au fur et à mesure dans des variables).

2°) Traduisez l'algorithme en une fonction Python.

3°) Utilisez cette fonction pour trouver une solution de chacune des équations suivantes :

a) $x^2 + 4x = 12$ b) $x^2 - 2x - 15 = 0$ c) $2x^2 - 5x - 3 = 0$

Remarque : l'algorithme d'Al-Khwarizmi ne donne qu'une solution alors qu'il y en a parfois deux.

En dehors de sa méthode de résolution des équations du second degré, c'est grâce à Al-Khwarizmi que nous utilisons le système décimal positionnel inventé par les indiens (eux-mêmes inspirés par les chinois ?) ainsi que les techniques algébriques utiles à la résolution d'équation.