




Lycée d'enseignement général et technologique international Victor Hugo COLOMIERS		
	<b>Langages et programmation</b>	
	<b>Evaluation</b>	
Nom : _____ Prénom : _____		Date : _____

# 1 Objectifs et Consignes

## 1.1 Objectifs

Au cours de cette évaluation, nous aurons pour objectif de programmer la fonction  $x^y$  de la calculatrice (x, y entiers relatifs). Afin d'y parvenir, nous progresserons au travers de plusieurs étapes.

## 1.2 Consignes

- Les programmes seront réalisés à l'aide de l'IDLE Python.
- Vous livrerez vos algorithmes libellés avec les noms de fichier indiqués dans le texte du document (ex : « deuxPuissance\_algo.py »).
- Vous livrerez vos programmes libellés avec les noms de fichier indiqués dans le texte du document (ex : « deuxPuissance.py »).
- Chaque programme sera réalisé en insérant directement les lignes de code sous les lignes de langage algorithmique qui apparaissent en commentaire (symbole #)

Exemple :

# AUTREMENT SI a<b ALORS

← ligne de langage algorithmique

elif a < b:

← ligne de code Python

- Une ligne de code est insérée sous une ligne de langage algorithmique



## Langages et programmation



### Evaluation

## 2 Les puissances de l'entier 2

1. Ouvrir le fichier « deuxPuissance\_algo.py »

2. Compléter l'algorithme ci-dessous :

```
# coding: utf-8
# nom du programme : deuxPuissance_algo.py
# rôle : calculer et afficher les puissances de l'entier 2

# Initialisation des variables et constantes
# exposant <-- 0
# ... code ...

# résultat <-- ?
# ... code ...

# Entrée clavier de l'exposant
# ... code ...

# Pour i allant de 0 à l'exposant exclu
# ... code ...

# ...?
# ... code ...

# Afficher le résultat (sortie écran)
# ... code ...
```

Sauvegardez votre fichier « deuxPuissance\_algo.py » car il sera à livrer dans une archive.

3. En partant de l'algorithme « deuxPuissance\_algo.py », insérer les lignes de code sous les lignes de langage algorithmique (ajoutées au 2.) et sauvegardez votre programme avec le nom de fichier « deuxPuissance.py ». A livrer.



## Langages et programmation

### Evaluation



## 3 Puissance signée de l'entier 2

1. Modifier l'algorithme « deuxPuissance\_algo.py » afin qu'il permette le calcul de  $2^y$  avec  $y$  entier relatif. Sauvegardez vos modifications avec le libellé « deuxpuissance\_signée\_algo.py ». Aide : Vous pourrez éventuellement vous inspirer de l'algorithme incomplet suivant :

```
# coding: utf-8
# nom du programme : deuxPuissance_signée_algo
# rôle : calculer et afficher les puissances de l'entier 2
# propriété : l'exposant est un entier relatif

# Initialisation des variables et constantes
# exposant <-- 0
# ... code ...

# résultat <-- ?
# ... code ...

# signe de l'exposant <-- ?. Par défaut, l'exposant est un entier naturel
# ... code ...

# Entrée clavier de l'exposant
# ... code ...

# Traitement du signe de l'exposant
# Si l'exposant est négatif
# ... code ...

# .?.
# ... code ...

# .?.
# ... code ...

# Pour i allant de 0 à l'exposant exclu
# ... code ...

# .?.
# ... code ...

# Afficher le résultat (sortie écran)
# .?.
# ... code ...

# .?.
# ... code ...

# .?.
# ... code ...

# .?.
# ... code ...
```

Sauvegardez votre fichier « deuxPuissance\_signée\_algo.py » car il sera à livrer dans une archive.



## Langages et programmation

### Evaluation



2. En partant du fichier « deuxPuissance\_signée\_algo.py », insérer les lignes de code sous les lignes de langage algorithmique (ajoutées au 1.) et sauvegardez votre programme avec le nom de fichier «deuxPuissance\_signée.py»
3. Testez l'exécution de votre programme. Le résultat doit être similaire à :

```
>>>
= RESTART: D:\Mes doc synchronisés\LIVH\NSI\Terminale\Langages et programmation\
évaluation\sources\puissance_2_signée.py
Entrez la valeur de l'exposant : 5
  2 puissance 5 = 32
>>>
= RESTART: D:\Mes doc synchronisés\LIVH\NSI\Terminale\Langages et programmation\
évaluation\sources\puissance_2_signée.py
Entrez la valeur de l'exposant : -5
  2 puissance -5 = 0.03125
```



## Langages et programmation

### Evaluation



## 4 Fonction 2<sup>y</sup>

1. Modifiez l'algorithme « deuxPuissance\_signée\_algo.py » afin qu'il permette le calcul de 2<sup>y</sup> avec y entier relatif et qu'il inclue la fonction deuxPuissance(). Cette fonction prend un entier relatif comme argument. Sauvegardez vos modifications avec le libellé «deuxPuissance\_signée\_fonction\_algo.py». Aide : Vous pourrez éventuellement vous inspirer de l'algorithme incomplet suivant :

```
# coding: utf-8
# nom du programme : deuxPuissance_signée_fonction_algo
# rôle : calculer et afficher les puissances de l'entier 2
# propriété : l'exposant est un entier relatif
#           le programme appelle la fonction deuxPuissance(exposant)

# Initialisation des variables et constantes
# exposant <-- 0
# ... code ...

# résultat <-- ?
# ... code ...

# signe de l'exposant <-- ?. Par défaut, l'exposant est un entier naturel
# ... code ...

# déclaration de la fonction deuxPuissance
# ... code ...

    # .?.
    # ... code ...

    # Pour i allant de 0 à l'exposant exclu
    # ... code ...

        # .?.
        # ... code ...

    # .?.
    # ... code ...

# Entrée clavier de l'exposant
# ... code ...

# Traitement du signe de l'exposant
# Si l'exposant est négatif
# ... code ...

    # .?.
    # ... code ...

    # .?.
    # ... code ...

# Afficher le résultat (sortie écran)
# .?.
# ... code ...

    # .?.
    # ... code ...

# .?.
# ... code ...

    # .?.
    # ... code ...
```

Sauvegardez votre fichier «deuxPuissance\_signée\_fonction\_algo.py» car il sera à livrer dans une archive.



## Langages et programmation

### Evaluation



2. En partant du fichier « deuxPuissance\_signée\_fonction\_algo.py », insérer les lignes de code sous les lignes de langage algorithmique (ajoutées au 1.) et sauvegardez votre travail avec le nom de fichier «deuxPuissance\_signée\_fonction.py»
3. Testez l'exécution de votre programme. Le résultat doit être similaire au résultat obtenu en exécutant le programme précédent « deuxPuissance\_signée.py »

## 5 Fonction $x^y$

1. Modifiez l'algorithme « deuxPuissance\_signée\_fonction\_algo.py » afin qu'il permette le calcul de  $x^y$ ,  $x$  et  $y$  étant des entiers relatifs. Votre algorithme doit inclure la fonction récursive nPuissance().

Vous veillerez à respecter la spécification située dans la première partie de l'algorithme.

Sauvegardez vos modifications avec le libellé «nPuissance\_signée\_fonction\_réursive\_algo.py».

Aide: Vous pourrez éventuellement vous inspirer de l'algorithme incomplet fourni « nPuissance\_signée\_fonction\_réursive\_algo.py »

2. En partant du fichier « deuxPuissance\_signée\_fonction\_réursive\_algo.py », insérer les lignes de code sous les lignes de langage algorithmique (ajoutées au 1.) et sauvegardez votre travail avec le nom de fichier «deuxPuissance\_signée\_fonction\_réursive.py»
3. Testez l'exécution de votre programme. Le résultat doit être similaire à :

Calcul de  $x$  puissance  $y$ ,  $x$  et  $y$  entiers relatifs

Entrez la valeur de l'entier  $x$  : 5  
Entrez la valeur de l'exposant  $y$  : 6

5 puissance 6 = 15625

Calcul de  $x$  puissance  $y$ ,  $x$  et  $y$  entiers relatifs

Entrez la valeur de l'entier  $x$  : 6  
Entrez la valeur de l'exposant  $y$  : -7

6 puissance -7 = 3.5722450845907635e-06

Calcul de  $x$  puissance  $y$ ,  $x$  et  $y$  entiers relatifs

Entrez la valeur de l'entier  $x$  : -3  
Entrez la valeur de l'exposant  $y$  : -7

-3 puissance -7 = -0.0004572473708276177



## Langages et programmation



### Evaluation

Calcul de x puissance y, x et y entiers relatifs

Entrez la valeur de l'entier x : 7

Entrez la valeur de l'exposant y : -3.2

Erreur de saisie : l'entier et l'exposant sont des entiers relatifs

Calcul de x puissance y, x et y entiers relatifs

Entrez la valeur de l'entier x : -4.15

Erreur de saisie : l'entier et l'exposant sont des entiers relatifs

Calcul de x puissance y, x et y entiers relatifs

Entrez la valeur de l'entier x : -4

Entrez la valeur de l'exposant y : 5

-4 puissance 5 = -1024

---

Liste des fichiers à livrer dans une archive libellée « nom\_evaluation\_TNSI\_LP » :

deuxPuissance  
deuxPuissance\_algo  
deuxPuissance\_signée  
deuxPuissance\_signée\_algo  
deuxPuissance\_signée\_fonction  
deuxPuissance\_signée\_fonction\_algo  
nPuissance\_signée\_fonction\_réursive  
nPuissance\_signée\_fonction\_réursive\_algo

Tous les fichiers possèdent l'extension « \*.py »

\*\*\*\* Fin du document \*\*\*\*