MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE

DIRECTION GENERALE DE L'ENSEIGNEMENT SUPERIEUR
ET DE L'INSERTION PROFESSIONNELLE (**DGESIP**)

Institut National Polytechnique
Félix Houphouët – Boigny
SERVICE DES CONCOURS

REPUBLIQUE DE CÔTE D'IVOIRE
Union - Discipline - Travail

Concours STIC/GIC session 2017

Composition : Informatique 1

Durée : 3 Heures

« Ordinateurs, tablette et autres téléphones interdits, Documents non autorisés » »
Si, au cours de l'épreuve, un candidat repère ce qui lui semble être une erreur d'énoncé, il le signale sur sa copie et poursuit sa composition en expliquant les raisons des initiatives qu'il est amené à prendre.

Exercice 1 Complexité algorithmique

1) Pour chacun des algorithmique suivant, indiquez la complexité au pire des cas de chaque étape de l'algorithme ainsi que la complexité de la fonction.

```
def boucle(x):
    x = x + 1
    i = 0
    while i < x:
        i = i + 1
    x = x * x
    return x
```

```
def devine(a,b) :
    if (a = = 0) :
        return b
    return devine(a-1,a*b)
```

```
def nesting(n):
    m = n
    sum = 0
    i = 0
    while i < n:
        j = 0
        while j < n:
            k = 0
            while k < m:
                sum += 2
                k += 1
            j += 1
        i += 1
    return sum
```

Exercice 2 : Codage en Python

Le développement limité de MAC LAURIN au voisinage de $x = 0$ à l'ordre n pour une fonction f infiniment dérivable s'écrit :

$$f(x) = f(0) + \frac{x}{1!}f'(0) + \frac{x^2}{2!}f''(0) + \cdots + \frac{x^n}{n!}f^{(n)}(0) + x^n\,\varepsilon(x) \quad \text{avec} \lim_{x\to 0}\varepsilon(x) = 0$$

On obtient au voisinage de 0 les développements limités suivants :

$$\frac{1}{1-x} = 1 + x + x^2 + \ldots + x^n + x^n\,\varepsilon(x)$$

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \cdots + \frac{x^n}{n!} + x^n\,\varepsilon(x)$$

$$\ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} + \cdots + (-1)^{n-1}\frac{x^n}{n} + x^n\,\varepsilon(x)$$

$$\text{Arctan(x)} = x - \frac{x^3}{3} + \frac{5}{5} + \cdots + (-1)^p\frac{x^{2p+1}}{2p+1} + x^{2p+2}\,\varepsilon(x) \quad (n = 2p+2)$$

Partie 1 : On désire écrire un algorithme Python qui calcule au voisinage de 0 la fonction $ln\,(1+x)$, et sa dérivée à un ordre ou rang k donné.

NB : Dans la suite, on pourra utiliser ou appeler une fonction déjà écrite dans une question précédente dans une autre question qui suit sans la réécrire à nouveau. Les fonctions sont écrites en pseudo-code.

1) Ecrire une fonction appelée ***factoriel*** qui permet de calculer le factoriel d'un entier n donné en paramètre.

2) Ecrire une fonction appelée ***puissance*** qui permet de calculer la puissance d'un réel a par un entier n (a^n) fournis en paramètre.
3) Ecrire une fonction qu'on appellera **Ln** qui prend en paramètre un réel x tel que $-1 < x < 1$, et un entier k puis calcule $ln(1+x)$ à l'ordre k (on ne tiendra pas compte de $x^n\,\varepsilon(x)$), c'est-à-dire $\sum_{n=1}^{k}(-1)^{n-1}\,(\frac{1}{n}x^n)$
4) La fonction ln est dérivable, et on a (ln U)'= U'/U. (ln U)' étant la dérivée de ln U. On a donc (ln (1+x))' = $\frac{1}{1+x}$. Aussi, on (ln(1-x))' = $-\frac{1}{1-x}$
   a) Ecrire une fonction qu'on appellera **deriveeLn** qui prend en paramètre un réel x tel que $-1 < x < 1$, et un entier k puis calcule la dérivée de $ln(1-x)$ à l'ordre k.
   b) Ecrire une fonction qu'on appellera **deriveeLn2** qui prend en paramètre un réel x tel que $-1 < x < 1$, et un entier k puis calcule la dérivée de $ln(1+x)$ à l'ordre k.

Partie 2 :
On désire écrire un algorithme qui calcule au voisinage de 0 la réciproque de la fonction ln(x), notée e^x à un ordre donné.

1) Ecrire une fonction qu'on appellera **exponentiel** qui prend en paramètre un réel x et un entier k, puis calcule e^x au voisinage de 0 à l'ordre k, c'est-à-dire $\sum_{n=0}^{k}\frac{1}{n!}x^n$ .On pourra utiliser les fonctions **puissance** et **factoriel** déjà existantes.
2) Ecrire un programme Python donnant la courbe de la fonction **exponentiel(x,5)** pour $-1 < x < 1$

Exercice 3 : Base de données

Un technicien propose le schéma de gestion de la base de données d'une compagnie ferroviaire. Les relations portant sur la partie gérant les trains ont été présentées ici. La description des 3 relations est la suivante :

TRAIN	TRAJET	TYPE
immatriculation : entier gare_attache : texte #id : entier	num_trajet : entier #immatriculation : entier ville_dep : texte ville_arr : texte heure_dep : date heure_arr : date	id : entier nom : texte nb_place : entier

Dans chaque relation, la clé primaire est soulignée. Pour les tables contenant une clé étrangère, celle-ci est préfixée par le caractère dièse (#).

1) Donnez la suite de commandes necessaires à lma création de cette base de données GESTION_TRAIN en se limitant aux 3 relations.
2) Ecrivez une requête SQL donnant comme résultat l'heure de départ, l'heure d'arrivée et l'identifiant de tous les trains au départ de la ville de KOMANDIMAN.
3) Modifiez la requête de la question 2 afin d'afficher, en plus des informations demandées, la gare d'attache de tous les trains toujours au départ de la ville de KOMANDIMAN. Attention on ne veut pas de doublons.
4) Ecrivez une requête SQL donnant comme résultat le nombre de trajets au départ de Toumodi par type de train.
5) Ecrivez une requête SQL donnant le nom et la gare de départ de tous les trains dont l'heure de départ est comprise entre 7h et 13h.
6) Le total des heures de formateurs depuis le 1er janvier 2017