EXAMEN : Algorithme et Programmation Université Paris-Dauphine MIDO L2

Pour les exercices demandant de compléter un programme python les parties manquantes sont représentées par `?(1)?`, `?(2)?`, `?(3)?`... Le format de votre réponse sera par exemple

```
(1)   a=b
(2)   return x
(3)   f(x,y)
```

où `a=b` est le contenu manquant à l'endroit `?(1)?`,
`return x` est le contenu manquant à l'endroit `?(2)?`,
`f(x,y)` est le contenu manquant à l'endroit `?(3)?`...

Exercice 1 *Combien vaut* $\lceil \log_3(221) \rceil$ *? justifiez.*

(1 point) 5, puisque $81 = 3^4 < 221 < 240 = 3 \times 80 < 3 \times 81 = 3^5$.

Exercice 2 *Combien d'opérations élémentaires (tests, affectations, opérations arithmétiques) effectuent ce programme en python ?*

```
def A():
     a=8
     while a > 0:
           for i in range(2):
                a = a - 1
```

(1 point) `A()` fait

- 1 affectation,
- 5 tests dans `while` (pour `a`=8,6,4,2,0)
- soit 4 passages dans `while` et donc $4 \times 8 = 32$ opérations sur `i` dans `for` (Puisque range(n) fait n+1 tests, n+1 affectations et n addition soit 3n+2=8 opérations) (la réponse avec seulement n tests soit 7 opérations pour range(2) au lieu de 8 est fausse mais tolérée)
- 8+8 opérations décrémentation de `a` (soustraction + affectation)

= 6+32+16=54 opérations élémentaires. (ou 50 toléré)

Exercice 3 *Complétez ce programme en python retournant le plus grand diviseur commun de deux entiers.*

```
def e(a,b):
      if a*b==0:
          return ?(1)?
      else:
          if a>b:
              return ?(2)?
          else:
              return ?(3)?
```

(1 point)

```
(1)  a+b
(2)  e(b,a%b)
(3)  e(a,b%a)
```

Exercice 4 *Complétez ce programme où a et b en entrée sont deux entiers tels que $a \geq b$ qui retourne un triplet d'entiers relatifs* `[d,u,v]` *tel que $au + bv = d =$pgcd(a, b).*

```
def B(a,b):
        if b==0:
                return [a,1,0]
        else:
                [d,u,v]=B(b,a%b)
                return [d,v,?(1)?]
```

(2 points)

```
(1) u-(a//b)*v
```

Exercice 5 *Complétez ce programme de tri fusion en python.*

```
def tf(t):
    if  len(t) <= 1:
        return t
    m = len(t)//2
    t1 = t[:m]
    t2 = t[m:]
    ?(1)?
    ?(2)?
    tt = fus(tt1,tt2)        #fus(tt1,tt2) retourne l'union triée de deux tableaux tt1,tt2 triés
    return tt
```

(1 point)

```
(1)     tt1 = tf(t1)
(2)     tt2 = tf(t2)
```

Exercice 6 *Complétez ce programme de tri insertion en python.*

```
def trins(A):
        for j in range(1,len(A)):
                ?(1)?
                i=j-1
                while i>=0 and A[i]>x:
                        ?(2)?
                        i -= 1
                A[i+1]=x
```

(2 points)

```
(1)    x=A[j]
(2)    A[i+1]=A[i]
```

Exercice 7 *Complétez ce programme de tri dénombrement en python.*

```
def triden(A):
        n=len(A)
        m=max(A)   #retourne la valeur maximum du tableau A
        B=[0 for i in range(n)]
        C=[0 for j in range(m+1)]
        for i in range(n):
                C[A[i]] += 1
        for j in range(1,m+1):
                C[j]=C[j-1]+C[j]
        for i in range(n):
                B[?(1)?]=A[i]
        return B
```

(2 points)

(1) `C[A[i]]-1`

Exercice 8 *Soit $c > 0$ donnez une fonction $f(n)$ telle que $\sum_{i=0}^{n} c^i = \Theta(f(n))$.*

(1 point)

$$f(n) = \begin{cases} 1 & \text{si } c < 1 \\ n & \text{si } c = 1 \\ c^n & \text{si } c > 1 \end{cases}$$

Exercice 9 *Soient a, b, c des entiers. Donnez une fonction $f(n)$, telle que $\sum_{i=0}^{n} a^i b^{c(p-i)} = \Theta(f(n))$.*

(2 points) On a

$$\sum_{i=0}^{n} a^i b^{c(p-i)} = b^{cp} \sum_{i=0}^{n} \left(\frac{a}{b^c}\right)^i$$

donc on peut prendre

$$f(n) = \begin{cases} b^{cp}\left(\frac{a}{b^c}\right)^n = \frac{a^n}{b^{c(n-p)}} & \text{si } a > b^c \\ nb^{cp} & \text{si } a = b^c \\ b^{cp} & \text{si } a < b^c \end{cases}$$

(NOTA BENE: D'après le cours, on sait que $T(n) = \sum_{i=0}^{p} a^i b^{c(p-i)}$ est la complexité d'un algorithme récursif tel que $T(n) = aT(n/b) + n^c$, où $n = b^p$ est la taille des données et a, b, c des constantes. On sait aussi que

$$T(n) = \sum_{i=0}^{p} a^i b^{c(p-i)} = \begin{cases} a^p \sum_{i=0}^{p} \left(\frac{b^c}{a}\right)^{p-i} = a^p \sum_{i=0}^{p} \left(\frac{b^c}{a}\right)^i \\ b^{cp} \sum_{i=0}^{p} \left(\frac{a}{b^c}\right)^i = n^c \sum_{i=0}^{p} \left(\frac{a}{b^c}\right)^i \end{cases} = \Theta \begin{cases} a^p & \text{si } a > b^c \\ pa^p = pn^c & \text{si } a = b^c \\ n^c & \text{si } a < b^c \end{cases}$$

En supposant que $n = b^p$, on aurait

$$f(n) = T(n) + \sum_{i=p+1}^{n} a^i b^{c(p-i)} = T(n) + \begin{cases} a^p \sum_{i=p+1}^{n} \left(\frac{b^c}{a}\right)^{p-i} = a^p \sum_{i=1}^{n-p} \left(\frac{b^c}{a}\right)^{-i} = a^p \sum_{i=1}^{n-p} \left(\frac{a}{b^c}\right)^i \\ b^{cp} \sum_{i=p+1}^{n} \left(\frac{a}{b^c}\right)^i = n^c \sum_{i=p+1}^{n} \left(\frac{a}{b^c}\right)^i = n^c \sum_{i=1}^{n-p} \left(\frac{a}{b^c}\right)^{p+i} \end{cases}$$

$$= T(n) + \Theta \begin{cases} a^p \left(\frac{a}{b^c}\right)^{n-p} & \text{si } a > b^c \\ (n-p)a^p = (n-p)n^c & \text{si } a = b^c \\ n^c & \text{si } a < b^c \end{cases}$$

Donc, une réponse équivalente tolérée:

$$f(n) = \begin{cases} \frac{a^n}{b^{c(n-p)}} & \text{si } a > b^c \\ na^p = n^{c+1} & \text{si } a = b^c \\ n^c & \text{si } a < b^c \end{cases}$$

)

Exercice 10 *Montrer que pour tout entier $b \geq 2$ et toute fonction $T : \mathbb{N} \to \mathbb{R}^+$ croissante, si il existe un réel $k > 0$ tel que $T(b^p) = \Theta(b^{pk})$ pour tout entier p, alors $T(n) = \Theta(n^k)$ pour tout entier n.*

(2 points) Pour tout entier n, il existe un entier p tel que $b^p \leq n < b^{p+1}$. Donc $b^{kp} \leq n^k < b^{k(p+1)}$, et on a alors deux constantes $c, d > 0$ telles que:

$$\frac{c}{b^k} n^k < \frac{c}{b^k} b^{k(p+1)} = c\, b^{pk} \leq T(b^p) \leq T(n) \leq T(b^{p+1}) \leq d\, b^{k(p+1)} = db^k\, b^{kp} \leq db^k\, n^k$$

et donc deux constantes $c' = \frac{c}{b^k}, d' = db^k > 0$ telles que $c'n^k \leq T(n) \leq d'n^k$.

Exercice 11 *Montrer comment calculer le produit* $(ac - bd) + i(ad + bc)$ *de deux nombres complexes* $x = a + ib$ *et* $y = c + id$ *en effectuant que trois produits de réels.*

(2 points) $p_1 = (a + b)(c + d)$, $p_2 = ac$, et $p_3 = bd$. On a: $ac - bd = p_2 - p_3$ et $ad + bc = p_1 - p_2 - p_3$.

Exercice 12 *Complétez ce programme python dont l'appel* `stable(0)` *affiche, sous forme de tableau en ligne, tous les vecteurs* $x \in \{0,1\}^5$ *tels que* $Ax \leq \mathbf{1}$ *avec*

$$A = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix} \quad \mathbf{1} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

```
n=5
X=[0 for i in range(n)]
G=??(1)??
m=len(G)
def stable(i):
        for j in range(2):
                X[i]=j
                if i==n-1:
                        t=1
                        for e in range(m):
                                if ?(2)?:
                                        t=0
                        if t==1:
                                print X
                else:
                        ?(3)?
```

(3 points)

```
(1)       [[0,1],[1,2],[2,3],[3,4],[4,0]]
(2)       X[G[e][0]]==1 and X[G[e][1]]==1
(3)       stable(i+1)
```