EXAMEN rattrapage : 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 (1 point) *Combien vaut* $\lceil \log_2(333) \rceil$ *? justifiez.*

Exercice 2 (2 points) *Sachant que l'on dispose d'une fonction* `fus(tt1,tt2)` *qui retourne l'union triée de deux tableaux* `tt1,tt2` *triés, 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:]
    tt1= ?(1)?
    tt2= ?(2)?
    return ?(3)?
```

Exercice 3 (1 point) *Complétez ce programme de tri insertion en python.*

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

Exercice 4 (2 points) *Complétez ce programme en python qui retourne une valeur approchée de* $\sqrt{x}$ *à* ε *près.*

```
def sqrt(x,eps):
        r=10.0
        while r*r <= ?(1)?  or ?(2)? <= r*r :
                r = (r + ?(3)? )/2
        return r
```

Exercice 5 (2 points) *Complétez ce programme, où* n *et* $b > 1$ *en entrée sont des entiers, qui retourne l'expression de* n *en base* b*.*

```
def D(n,b):
    tab=[]
    while n>0:
        tab=?(1)?
        ?(2)?
    return tab
```

Exercice 6 (2 points) *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]=?(1)?
        for i in range(n):
                B[C[A[i]]-1]=A[i]
        return B
```

Exercice 7 (4 points) *Complétez ce code python pour qu'il affiche l'expression en base 5 de tous les nombres de 0 à* 5^5-1.

```
b=?(1)?
n=?(2)?
A=[0 for i in range(n)]
def count(i):
        for j in range(b):
                ?(3)?
                if ?(4)? :
                        print A
                else:
                        count(i+1)
?(5)?
```

Exercice 8 (6 points) *Soit n un entier. Les fonctions $g(x)=2x$ et $d(x)=2x+1$ permettent de structurer la représentation de $T=\{1,2,\ldots,n\}$ en tas binaire, la fonction $p(x)=\lfloor\frac{x}{2}\rfloor$ est alors telle que $p(g(x))=p(d(x))=x$. La fonction $h(x)$ associe un entier à chaque élément de T de la manière suivante: $h(1)=0$, et $h(x)=h(p(x))+1$ pour $x\geq 2$. On note $n(h)$ le nombre d'éléments x de T tels que $h(x)=h$. Montrez que*

$$n(h)\leq\left\lfloor\frac{n}{2^{h+1}}\right\rfloor$$