



Ens. : O. Lévêque, M. Stojilovic  
Information, Calcul, Communication - A  
Vendredi 1er novembre 2024  
Durée : 180 minutes

## U.N.Owen

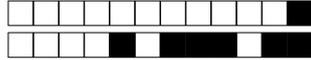
SCIPER: 0

Salle: INF 1

Attendez le début de l'épreuve avant de tourner la page. Ce document est imprimé recto-verso, il contient 12 pages, les dernières pouvant être vides. Ne pas dégrafer.

- Posez votre carte d'étudiant sur la table.
- Document autorisé pour cet examen : un formulaire constitué d'une page A4 recto-verso, manuscrite, préparée avec stylet+tablette ou à l'ordinateur.
- L'utilisation de tout appareil électronique (calculatrice, ordinateur, smartphone/watch, tablette) est interdite pendant l'épreuve.
- L'examen est composé de deux parties:
  - une partie avec 16 questions à choix multiple ; chaque question admet une seule réponse correcte : la réponse correcte vaut 2 points ; toute autre option (pas de réponse, réponse fausse, ou plusieurs cases cochées) vaut 0 point.
  - une partie avec des questions de type ouvert, valant en tout 24 points.
- Merci d'avance de soigner la présentation de vos réponses !
- Si une question est erronée, les enseignants se réservent le droit de l'annuler.

Respectez les consignes suivantes   Observe this guidelines   Beachten Sie bitte die unten stehenden Richtlinien		
choisir une réponse   select an answer Antwort auswählen	ne PAS choisir une réponse   NOT select an answer NICHT Antwort auswählen	Corriger une réponse   Correct an answer Antwort korrigieren
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> <input checked="" type="checkbox"/>
ce qu'il ne faut <b>PAS</b> faire   what should <b>NOT</b> be done   was man <b>NICHT</b> tun sollte		
<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>		



## Première partie, questions à choix multiple

Pour chaque question, marquer la case correspondante à la réponse correcte sans faire de ratures. Il n'y a qu'une seule réponse correcte par question.

### Question 1

On considère l'algorithme suivant:

<b>algo1</b>
entrée : <i>nombre entier positif</i> $n$ sortie : <i>nombre entier positif</i>
$s \leftarrow 0$ <b>Pour</b> $i$ allant de 1 à $n$ $j \leftarrow 1$ <b>Tant que</b> $j$ n'est ni un multiple de 5 ni un multiple de 7 $j \leftarrow j + 1$ $s \leftarrow s + j$ $j \leftarrow j + 1$ <b>Sortir</b> : $s$

Laquelle des affirmations suivantes est-elle correcte?

- Pour  $n \geq 1$  en entrée, la sortie de **algo1**( $n$ ) vaut  $5 \cdot n$ , et sa complexité temporelle est un  $\Theta(n^2)$ .
- Pour  $n \geq 1$  en entrée, la sortie de **algo1**( $n$ ) vaut  $5 \cdot n$ , et sa complexité temporelle est un  $\Theta(n)$ .
- Pour  $n \geq 1$  en entrée, la sortie de **algo1**( $n$ ) vaut la somme des  $n$  premiers multiples de 5 et de 7, et sa complexité temporelle est un  $\Theta(n)$ .
- Pour  $n \geq 1$  en entrée, la sortie de **algo1**( $n$ ) vaut la somme des  $n$  premiers multiples de 5 et de 7, et sa complexité temporelle est un  $\Theta(n^2)$ .

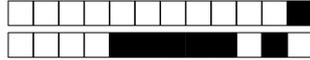
### Question 2

On considère l'algorithme suivant:

<b>algo2</b>
entrée : <i>liste</i> $L$ de <i>nombre réels strictement positifs</i> , de <i>taille</i> $n$ sortie : <i>nombre réel strictement positif</i>
$m \leftarrow 0$ <b>Pour</b> $i$ allant de 1 à $n$ $m \leftarrow m + \frac{1}{L(i)}$ <b>Sortir</b> : $n/m$

Quelle est la sortie de **algo2**( $L, n$ ) (laissée ici sous forme de fraction) si  $L = (2, 6, 6)$  et  $n = 3$  en entrée?

- $\frac{15}{6}$
- $\frac{18}{5}$
- $\frac{14}{3}$
- $\frac{24}{5}$

**Question 3**

On considère les deux algorithmes suivants:

<b>algo3</b>
entrée : <i>nombre entier positif</i> $n$ sortie : <i>nombre entier positif</i>
$\text{Si } n = 1$   <b>Sortir</b> : 1 <b>Sortir</b> : $\text{algo3}(n-1) + 2$

<b>algo3bis</b>
entrée : <i>nombre entier positif</i> $n$ sortie : <i>nombre entier positif</i>
$\text{Si } n = 1$   <b>Sortir</b> : 1 <b>Sortir</b> : $\text{algo3bis}(n-1) + \text{algo3}(n)$

Laquelle des affirmations suivantes est-elle correcte?

- $\text{algo3}(5) = 6$  et  $\text{algo3bis}(5) = 10$         $\text{algo3bis}(5)$  ne termine jamais  
  $\text{algo3}(5) = 9$  et  $\text{algo3bis}(5) = 25$         $\text{algo3}(5) = 11$  et  $\text{algo3bis}(5) = 36$

**Question 4**

On considère l'algorithme suivant:

<b>algo4</b>
entrée : <i>nombre entier positif</i> $n$ sortie : <i>nombre entier positif</i>
$\text{Si } n < 10$   <b>Sortir</b> : $n$ $m \leftarrow n \bmod 10$ <b>Sortir</b> : $m + \text{algo4}(\frac{n-m}{10})$

Laquelle des affirmations suivantes est-elle *incorrecte*?

- Si le nombre  $n$  en entrée est un multiple de 3, alors la sortie de  $\text{algo4}(n)$  est aussi un multiple de 3.  
 Si le nombre  $n$  en entrée est un multiple de 10, alors la sortie de  $\text{algo4}(n)$  est égale à la sortie de  $\text{algo4}(\frac{n}{10}) + 1$ .  
 La sortie de  $\text{algo4}(n)$  est la somme des chiffres qui composent le nombre  $n$ .  
  $\text{algo4}(\text{algo4}(\text{algo4}(5'347))) = 1$

**Question 5**

Quelle est la complexité temporelle de l'algorithme  $\text{algo4}(n)$  ci-dessus?

- $\Theta(n)$         $\Theta(10^n)$         $\Theta(1)$ , car c'est un        $\Theta(\log_{10}(n))$   
algorithme récursif

**Question 6**

Un enseignant d'une classe de 150 élèves (on suppose ici qu'il possède la liste des noms des élèves), veut enregistrer, pour chaque élève, de quelle section fait partie l'élève (GC ou MX) et aussi si cet élève a participé ou non au midterm du 1er novembre. Combien de bits sont-ils nécessaires au minimum pour enregistrer ces informations?

- 150       600       300       16



### Question 7

Laquelle des affirmations suivantes est-elle correcte?

- Tant qu'on n'aura pas montré que  $P = NP$ , on pourra utiliser tranquillement sa carte de crédit sur internet avec un protocole de sécurité reposant sur l'hypothèse qu'il n'est pas possible de factoriser des grands nombres en temps polynomial.
- On connaît aujourd'hui un algorithme qui permet de résoudre en temps polynomial le problème des sommes de sous-ensembles.
- Si on montre un jour que  $P \neq NP$ , alors on saura qu'il n'est pas possible de factoriser des grands nombres en temps polynomial.
- Si on montre un jour que  $P \neq NP$ , alors on aura montré qu'il n'existe pas d'algorithme permettant de résoudre le problème des sommes de sous-ensembles en temps polynomial.

### Question 8

On considère le problème suivant:

Etant donné une liste  $L$  de  $n$  nombres entiers relatifs,  
identifier le sous-ensemble  $S \subset \{1, \dots, n\}$  tel que  $|\sum_{i \in S} L(i)|$  soit minimale.

Laquelle des affirmations suivantes est-elle correcte?

- Ce problème fait partie de la classe P.
- Ce problème fait partie de la classe NP, mais on ne sait pas s'il fait partie de la classe P.
- Ce problème fait partie de la classe NP, donc il ne fait pas partie de la classe P.
- On ne sait pas si ce problème fait partie de la classe P, ni s'il fait partie de la classe NP.

### Question 9 Qu'affiche ce programme ?

```
x = 13
z = -8
y = z + 21

a = x == y
b = x == y - z
c = x > True > z

if a and b and c:
    if x - y != False:
        print("banana")
    else:
        print("ananas")
elif a or b or c:
    if x - y:
        print("penguin")
    else:
        print("unicorn")
else:
    print("pizza")
```

- ananas
- penguin
- banana
- unicorn
- pizza

**Question 10** Qu'affiche ce programme ?

```
numbers = [3, 6, 9, 12, 15]
result = 0

for i, v in enumerate(numbers):
    if i % 2 == 0:
        result += v
    elif i % 3 == 0:
        result -= v

print(result)
```

 15 3 9 -9 12**Question 11** Qu'affiche ce programme ?

```
matrix = [[i+2 * j for j in range(4)] for i in range(4)]
print(matrix)
```

 [[0, 2, 4, 6], [1, 3, 5, 7], [2, 4, 6, 8], [3, 5, 7, 9]] [[0, 1, 2, 3], [2, 3, 4, 5], [4, 5, 6, 7], [6, 7, 8, 9]] [[3, 5, 7, 9], [4, 6, 8, 10], [5, 7, 9, 11], [6, 8, 10, 12]] [[0, 2, 4, 6], [0, 3, 6, 9], [0, 4, 8, 12], [0, 5, 10, 15]] [[0, 0, 0, 0], [2, 3, 4, 5], [4, 6, 8, 10], [6, 9, 12, 15]]**Question 12** Qu'affiche ce programme ?

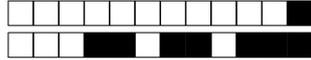
```
matrix = [[1, 0, 3, 4], [5, 6, 0, 8], [9, 10, 11, 0], [0, 14, 15, 16]]
result = 0
i = 0

while i < len(matrix):
    if i:
        for j in range(len(matrix[i])):
            if matrix[i][j] == 0:
                result += matrix[j][i]
        i += 1
    i += 1
print(result)
```

 29 Ce programme n'affiche rien car la boucle while ne se termine jamais. 14 34 20**Question 13** Sachant que la fonction intégrée `sum` de Python calcule la somme des éléments d'une liste donnée, que va afficher ce programme ? Exemple : `sum([1, 0, -5]) = -4`.

```
a = [i // 2 for i in range(1, 13, 3)]
b = [i * 2 for i in a]
a.extend(b)
result = sum(a)
for i, v in enumerate(b):
    result += v + b[i]
print(result)
```

 112 70 77.0 112.0 50



**Question 14** Sachant que les fonctions Python intégrées `min` et `max` trouvent respectivement la valeur minimale et maximale parmi les valeurs fournies, quelle est la valeur de la variable `result` à la fin de l'exécution de ce code ? Exemples : `min(1, -4, 9.2, -1) = -4`; `max(0, 3.3, -1, 6) = 6`.

```
x = c if (c > b) else b
x = a if (a > b and a > c) else x
y = min(-a, -b)
result = min(y, -c)
result -= x
```

- `max(a, b, c) - min(-a, -b, -c)`  
 `min(-a, -b, -c) - min(-a, -b, -c)`  
 `max(a, b, c) + min(a, b, c)`  
 `min(a, b, c) - max(-a, -b, -c)`  
 `2*min(-a, -b, -c)`

**Question 15** Sachant que la fonction intégrée `sum` de Python calcule la somme des éléments d'une liste donnée, que va afficher ce programme ? Exemple : `sum([1, 0, -5]) = -4`.

```
numbers = [0, 1, 2, 3, 4, 5, 6]
a = [x**2 for x in numbers]
b = [x and numbers[0] for x in numbers]
a.extend(b[-2:])
print(sum(a[1:3]), a[-1])
```

- 65 0       257 False       257 0       65 216       65 False

**Question 16** Sachant que `break` et `continue` affectent la boucle la plus proche, que va afficher ce programme ?

```
count = 0
for i in range(3):
    for j in range(3):
        if i == 1 and j == 0:
            count += 1
            break
        if i == j:
            continue
        count += 1
print(count)
```

- 3       7       4       5       10

## Deuxième partie, questions de type ouvert

Répondre dans l'espace dédié. Votre réponse doit être soigneusement justifiée, toutes les étapes de votre raisonnement doivent figurer dans votre réponse. Laisser libres les cases à cocher : elles sont réservées au correcteur.



**Question 17:** Cette question est notée sur 8 points.



On considère l'algorithme suivant:

<b>algo</b>
entrée : liste $L$ de nombres entiers positifs, de taille $n$ sortie : ???
$\text{Si } n = 1$   <b>Sortir:</b> $L(1)$ $m \leftarrow \lfloor \frac{n}{2} \rfloor$ <b>Sortir:</b> $\frac{m}{n} \cdot \text{algo}(L(1 : m), m) + \frac{n-m}{n} \cdot \text{algo}(L(m+1 : n), n-m)$

*Note:*  $L(a : b) = (L(a), L(a+1), \dots, L(b-1), L(b))$  désigne la sous-liste de  $L$  composée des éléments compris entre l'indice  $a$  et l'indice  $b$  (compris).

a) (1 point) Que vaut la sortie de l'algorithme  $\text{algo}(L, n)$  si  $L = (8, 10, 5, 9)$ ,  $n = 4$  en entrée? **Réponse:** 8

b) (1 point) Que vaut la sortie de l'algorithme  $\text{algo}(L, n)$  pour une liste  $L$  de taille  $n$  en général?  
**Réponse:** La sortie est égale à la moyenne des valeurs de  $L$ .

c) (2 points) Quelle est la complexité temporelle de  $\text{algo}(L, n)$ ? (utiliser la notation  $\Theta(\cdot)$  et justifier votre réponse; attention, car la réponse à la question n'est pas immédiate!)  
*Note:* On considère ici que l'opération  $\frac{m}{n} \cdot a + \frac{n-m}{n} \cdot b$  pour deux nombres  $a, b$  arbitraires compte comme une seule opération.

**Réponse:** La complexité temporelle est un  $\Theta(n)$ ; en effet,  $n - 1$  additions seront effectuées au total.

Supposons maintenant qu'on modifie l'algorithme de la page précédente comme suit:

<b>algobis</b>
entrée : liste $L$ de nombres entiers positifs, de taille $n$ sortie : ???
$\text{Si } n = 1$   <b>Sortir:</b> $L(1)$ $m \leftarrow n - 1$ <b>Sortir:</b> $\frac{m}{n} \cdot \text{algobis}(L(1 : m), m) + \frac{n-m}{n} \cdot \text{algobis}(L(m+1 : n), n-m)$

d) (2 points) Que vaut la sortie de ce nouvel algorithme  $\text{algobis}(L, n)$  en général? Justifier votre réponse.  
**Réponse:** La sortie ne change pas: c'est toujours la moyenne des valeurs de  $L$ .

e) (2 points) Quelle est la complexité temporelle de ce nouvel algorithme  $\text{algobis}(L, n)$ ? (utiliser à nouveau la notation  $\Theta(\cdot)$  et justifier votre réponse)  
**Réponse:** La complexité temporelle ne change pas non plus: c'est toujours un  $\Theta(n)$ , car à nouveau,  $n - 1$  additions sont effectuées au total.



**Question 18:** Cette question est notée sur 8 points.

<input type="checkbox"/>	0	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5	<input type="checkbox"/>	6	<input type="checkbox"/>	7	<input checked="" type="checkbox"/>	8
--------------------------	---	--------------------------	---	--------------------------	---	--------------------------	---	--------------------------	---	--------------------------	---	--------------------------	---	--------------------------	---	-------------------------------------	---

a) (4 points) Ecrire un algorithme qui prenne en entrée une liste  $L$  de nombres entiers relatifs, de taille  $n$ , dont la sortie soit oui si et seulement si

$$\sum_{i=1}^k L(i) \geq 0, \quad \forall 1 \leq k \leq n \quad (1)$$

et dont la complexité temporelle soit un  $\Theta(n)$ .

**Réponse:**

<b>algo1</b>
entrée : liste $L$ de nombres entiers relatifs, de taille $n$ sortie : oui/non
<pre>s ← 0 <b>Pour</b> i allant de 1 à n     s ← s + L(i)     <b>Si</b> s &lt; 0       <b>Sortir</b> : non <b>Sortir</b> : oui</pre>

b) (4 points) On considère maintenant l'algorithme suivant:

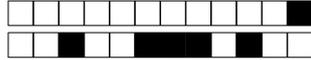
<b>algo</b>
entrée : liste $L$ de nombres entiers relatifs, de taille $n$ sortie : oui/non
<pre><b>Pour</b> i allant de 1 à n     <b>Si</b> L(i) &lt; 0       <b>Sortir</b> : non <b>Sortir</b> : oui</pre>

Ecrire un algorithme dont les entrées et sortie soient identiques à celui de la partie a), mais qui utilise explicitement l'algorithme **algo** ci-dessus comme sous-algorithme.

*Note:* Bien sûr, votre algorithme fera appel à **algo** avec une liste  $L$  différente de celle pour laquelle il cherche à savoir si la relation (1) de la page précédente est vérifiée ou non.

**Réponse:**

<b>algo2</b>
entrée : liste $L$ de nombres entiers relatifs, de taille $n$ sortie : oui/non
<pre>M ← (L(1)) <b>Pour</b> i allant de 2 à n     M ← M + (M(i-1) + L(i)) (= "ajouter l'élément M(i-1)+L(i) à la liste M") <b>Sortir</b> : algo(M, n)</pre>



+1/9/52+

**Question 19:** Cette question est notée sur 8 points.

<sub>0</sub>
<sub>1</sub>
<sub>2</sub>
<sub>3</sub>
<sub>4</sub>
<sub>5</sub>
<sub>6</sub>
<sub>7</sub>
<sub>8</sub>

**a) (1 point)** Soit  $x$  le nombre entier positif représenté en binaire par 10010000. Quelle est la représentation binaire du nombre  $x/2$ ?

**Réponse:** 1001000

**b) (1 point)** Quelle est la représentation binaire du nombre  $\sqrt{x}$ ?

**Réponse:** 1100

**c) (1 point)** En général, si  $x$  est un nombre entier positif pair représenté sur  $n$  bits, combien de bits seront-ils nécessaires pour représenter le nombre  $x/2$ ?

**Réponse:**  $n - 1$

**d) (1 point)** En général, si  $x$  est un nombre entier positif représenté sur  $n$  bits, et  $x = y^2$  avec  $y$  un nombre entier positif, combien de bits (approximativement) seront-ils nécessaires pour représenter le nombre  $y$ ?

**Réponse:** environ  $n/2$

**e) (1 point)** En considérant maintenant la représentation binaire des nombres entiers *relatifs* sur 8 bits, quel est le nombre représenté par 10010000?

**Réponse:** -112

**f) (2 points)** Dans cette même représentation, est-ce que l'opération

$$\begin{array}{r}
 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0 \\
 +\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0 \\
 \hline
 \end{array}$$

mène à un overflow? Justifiez votre réponse.

**Réponse:** Oui, la somme vaut -224, qui est en dehors du domaine couvert par cette représentation.

**g) (1 point)** Toujours dans cette même représentation, quel est le plus grand nombre entier positif qu'il est possible de représenter? (donner sa représentation décimale)

**Réponse:** +127