

**[CYBER1][2024-2025] Examen (Sujet B) (1h30)**  
 Algorithmique 1

NOM :

PRÉNOM :

Vous devez respecter les consignes suivantes, sous peine de 0 :

- Lisez le sujet en entier avec attention
- Répondez sur le sujet
- Ne détachez pas les agrafes du sujet
- Écrivez lisiblement vos réponses (si nécessaire en majuscules)
- Ne trichez pas

## 1 Questions (6 points)

1.1 (2 points) Sélectionnez les conditions vraies pour  $A = 6$  et  $B = 2$  :

- ☐  $((A + 2) \geq (B + 6))$  et  $(B < A)$   
☐  $((A - 2) \geq (B + 2))$  et  $(A < B)$   
☐  $((A + B) \leq 9)$  et  $((6 - B) < 4)$   
☐  $((B + 3) > (A - 1))$  ou  $(B == (A - 4))$

1.2 (4 points) Exécutez cet algorithme avec les valeurs  $(x = 9)$   $(y = 3)$  et  $(z = 3)$  en remplissant le tableau, puis donnez les caractéristiques de cet algorithme :

```

algorithme fonction Calc : entier
  parametres locaux
    entier    x, y, z
  debut
    si (x <= 1)
      retourne (1)
    sinon
      si ((x % z) == 0)
        retourne (y + Calc((x - y), (y + 1), z))
      sinon
        retourne (Calc((x - y), (y - 1), z))
      fin si
    fin si
  fin algorithme fonction Calc
  
```

tour	x	y	z
État Initial			
Total :			

- ☐ Il est récursif  
☐ Il est même récursif terminal  
☐ Il s'agit d'une fonction  
☐ Il s'agit d'une procédure

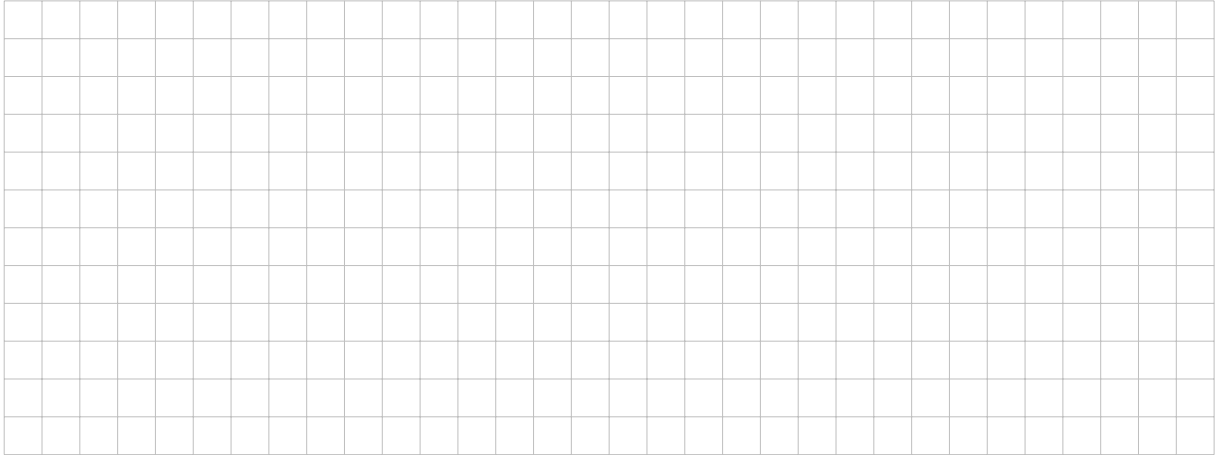
## 2 Algorithmes (14 points)

- 2.1 (1 point) Écrivez une fonction « *Facto* » récursive calculant la factorielle du nombre N. (Vous décrirez d'abord le ou les cas d'arrêts, puis le cas général)

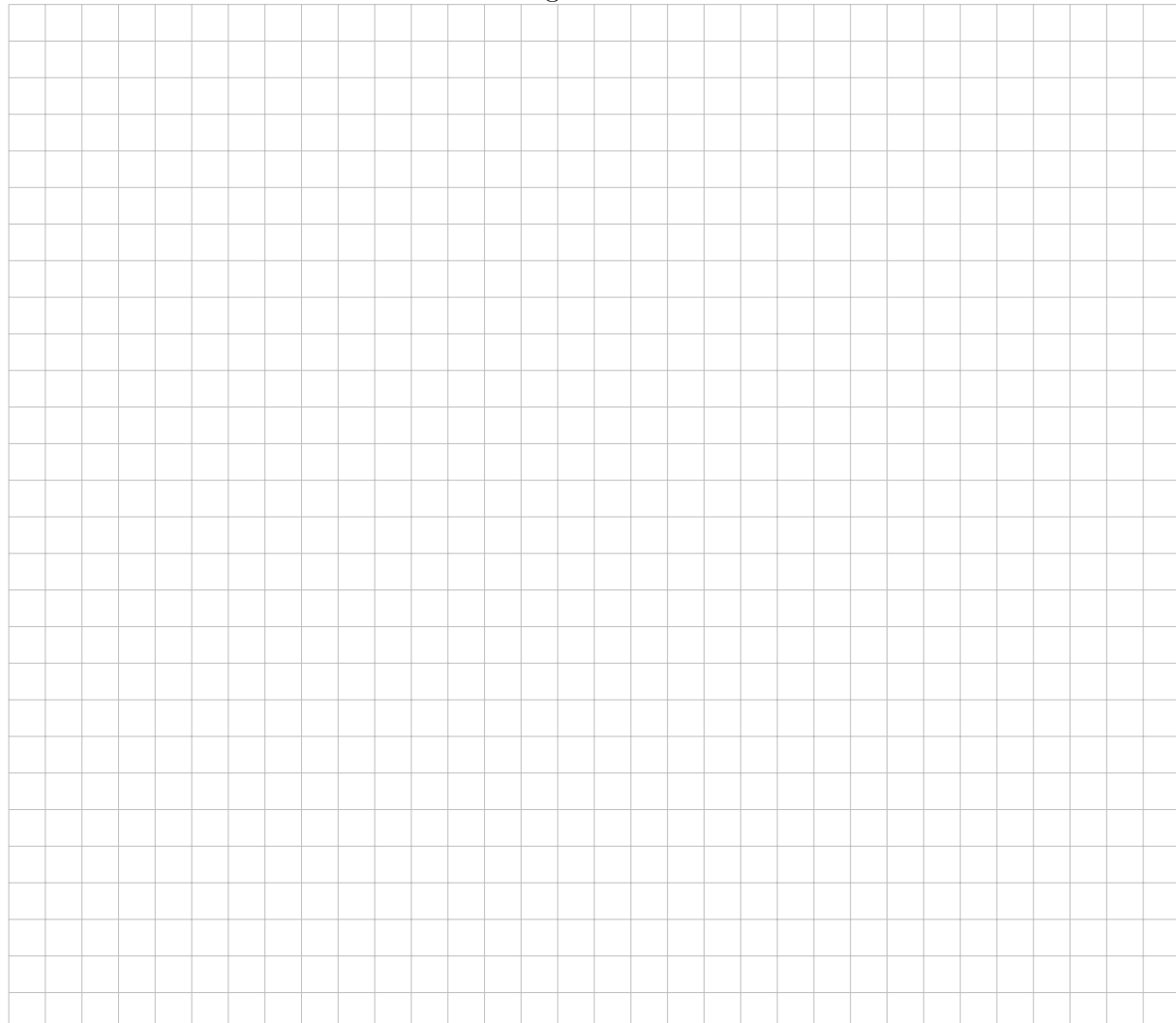
$$n! = n \times (n - 1) \times \dots \times 1$$

$$0! = 1! = 1$$

Explications du/des cas d'arrêts, puis du cas général :



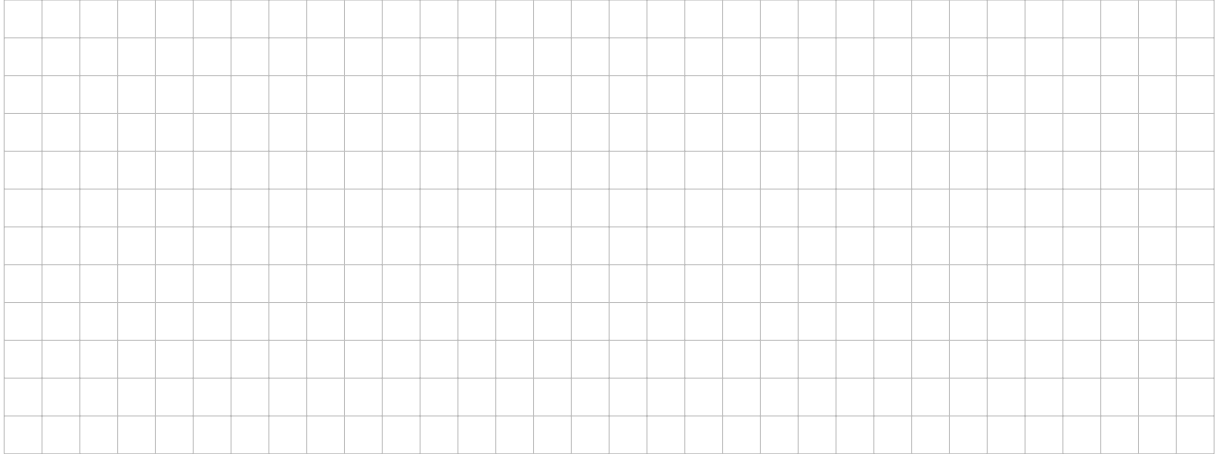
Algorithme :



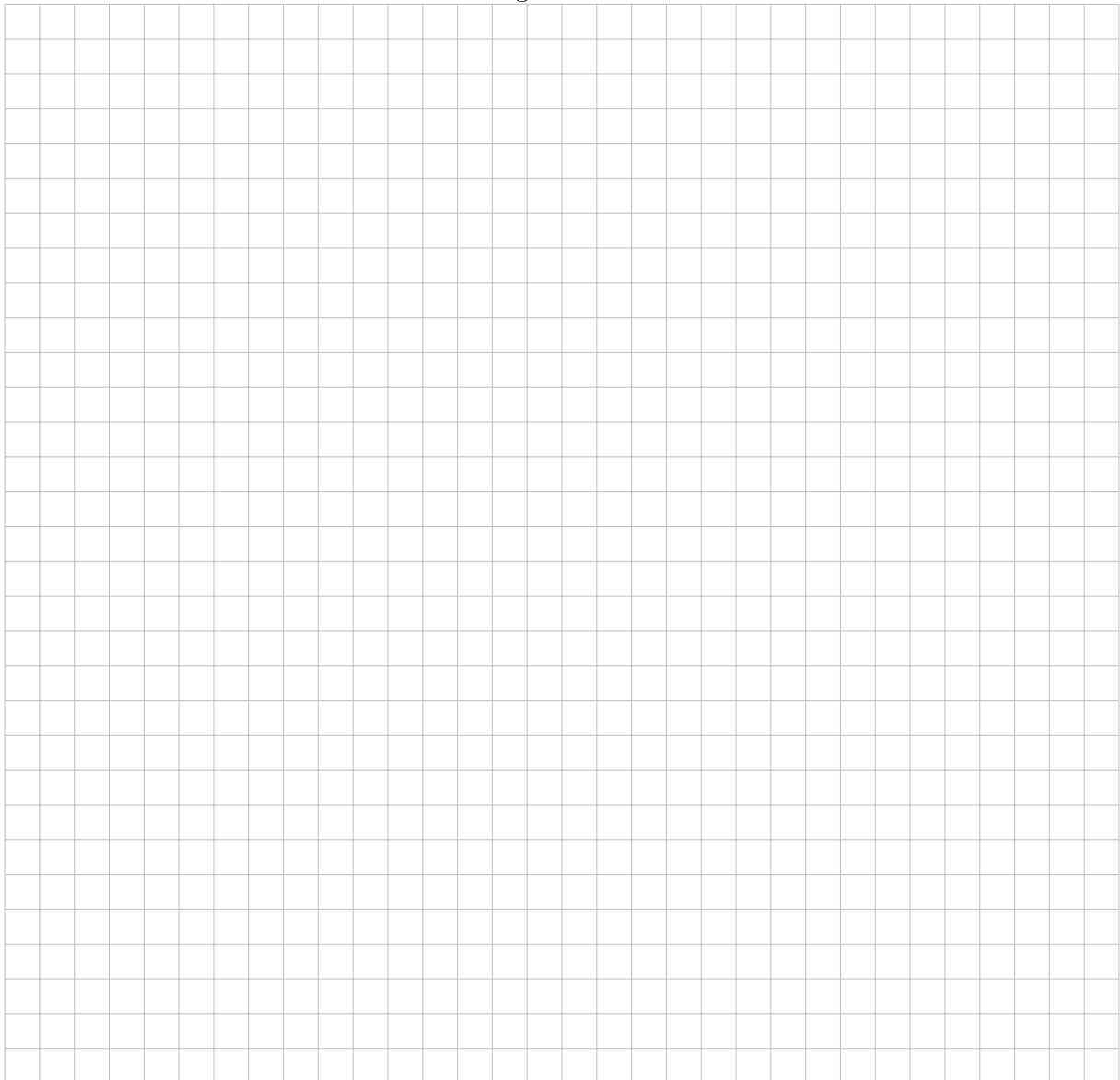
- 2.2 (2 points)** Écrivez une fonction « *SommePairs* » itérative calculant la somme des N premiers nombres pairs. (Vous décrirez d'abord le ou les cas d'arrêts, puis le cas général)

$$SommePairs(4) = 6 + 4 + 2 + 0 = 12$$

Explications du/des cas d'arrêts, puis du cas général :



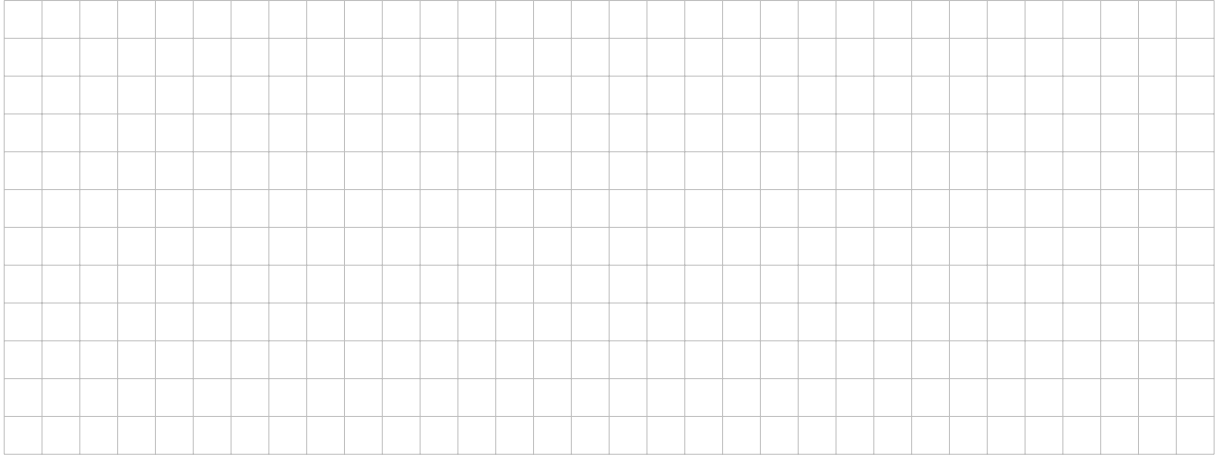
Algorithme :



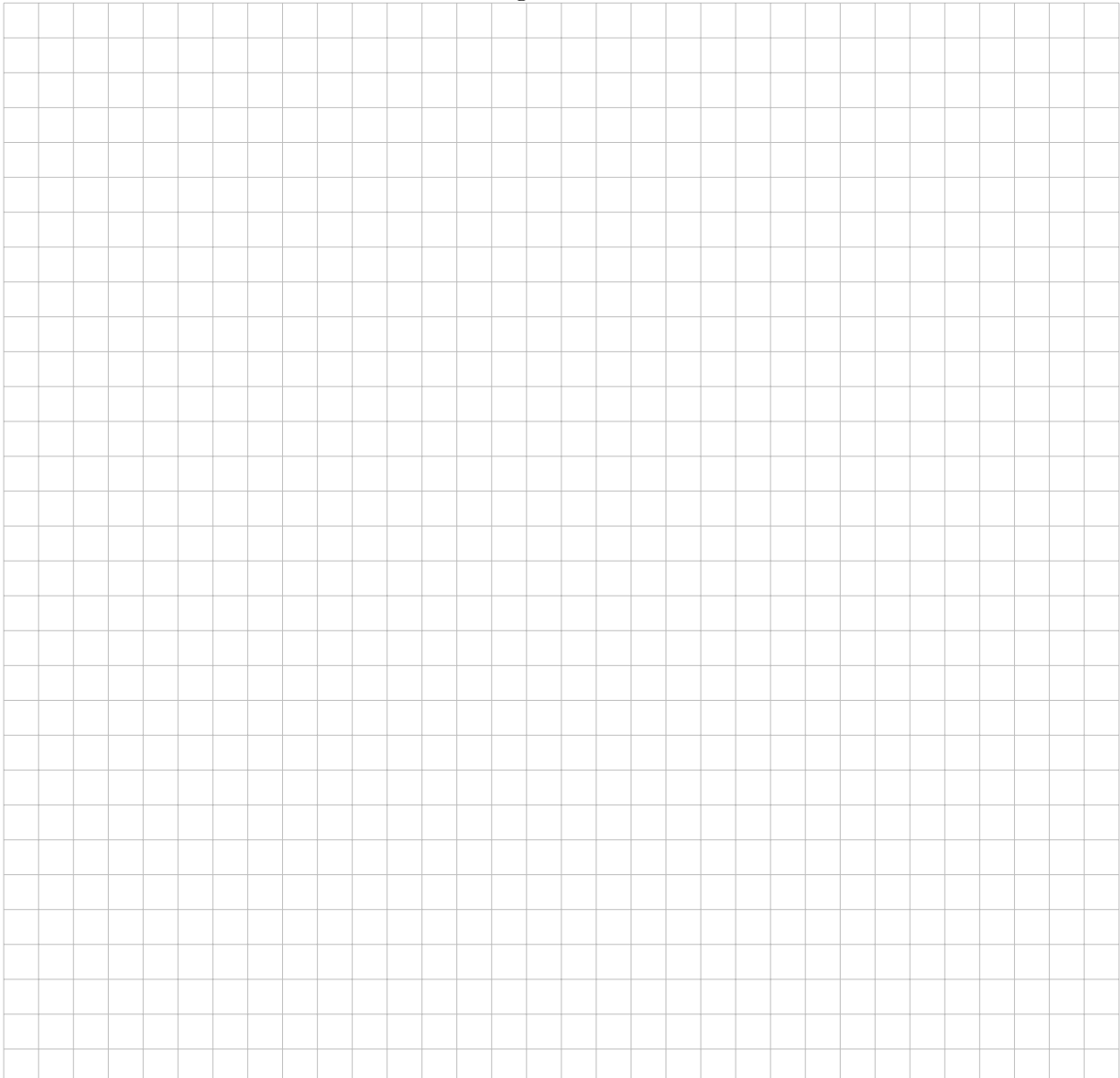
**2.3 (2 points)** Écrivez une fonction « *strlen* » itérative calculant la taille d'une chaîne de caractères. (Vous décrierez d'abord le fonctionnement général)

$$\text{strlen}(\text{"Dino"}) = 4$$

Explications du fonctionnement :



Algorithme :

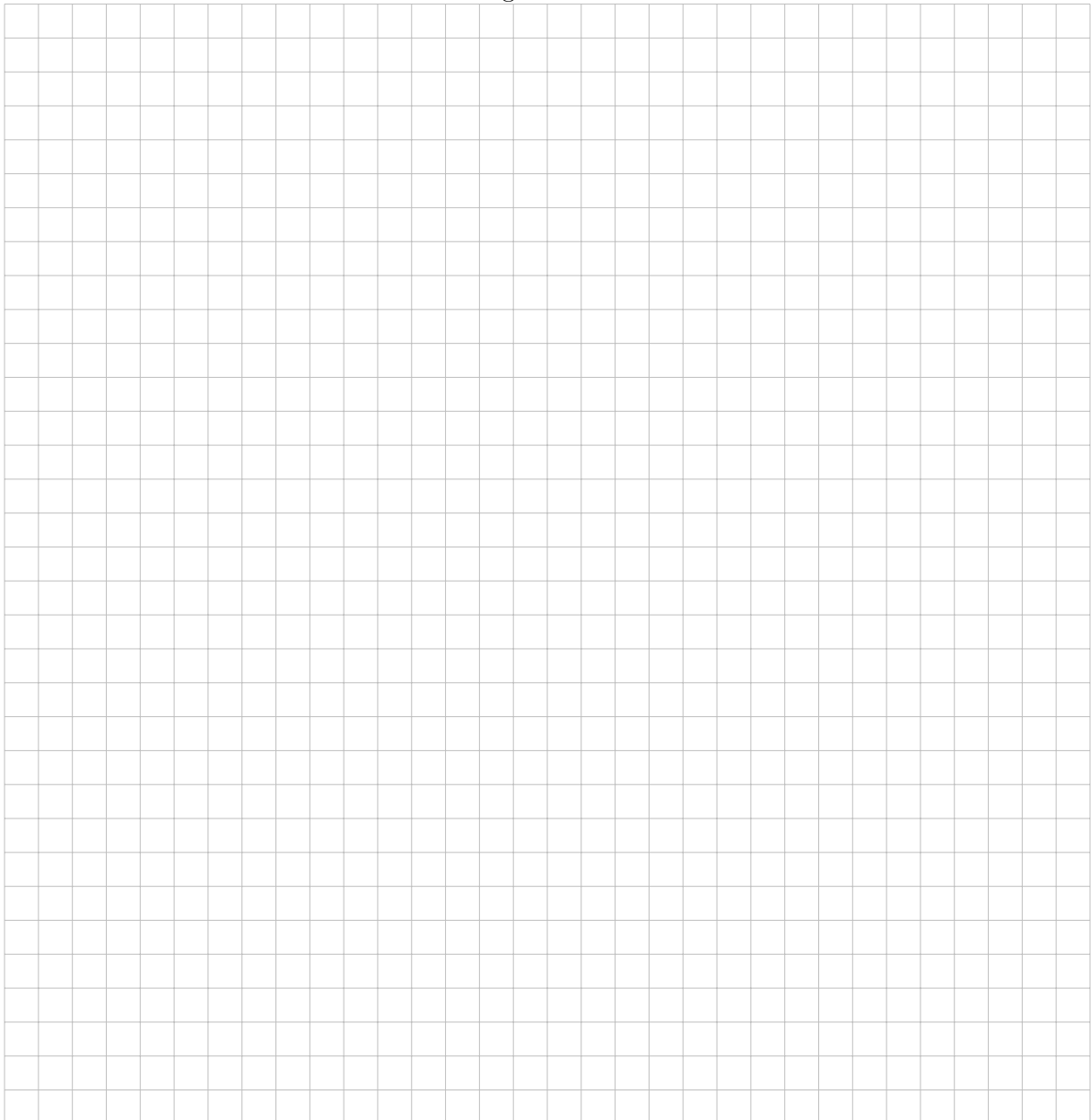


- 2.4 (2 points) Écrivez une fonction « *RechercheMinTab* » cherchant l'emplacement de l'élément le plus petit dans un tableau d'entiers. Si tous les éléments sont les mêmes, vous renverrez la position de votre choix du tableau (tant que cette position existe). (Vous décrirez d'abord le fonctionnement général)

Explications du fonctionnement :




Algorithme :



**2.5 (3 points)** Écrivez une fonction « *AfficheVoy* » affichant chaque voyelle contenue dans une chaîne de caractères, et renvoyant le nombre de voyelles trouvées. Si la chaîne n'en contient aucune, vous n'afficherez rien et vous renverrez 0.

*Vous pourrez utiliser la fonction **IsVoyelle** qui prend un caractère en paramètre et renvoie si oui ou non il s'agit d'une voyelle, ainsi que **Print** qui affiche le paramètre qui lui est donné.*

Algorithme :

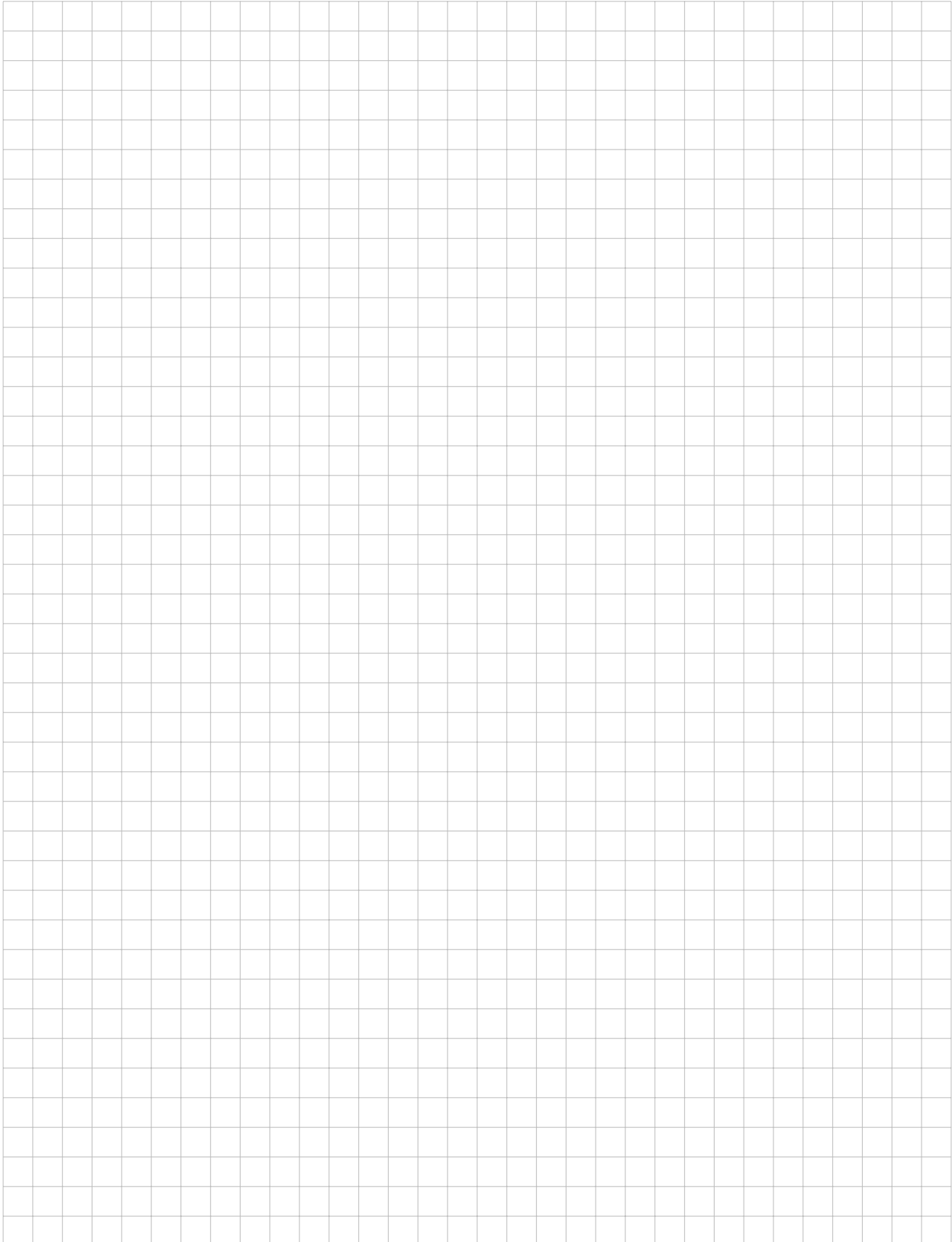
A large grid of graph paper, consisting of 30 columns and 40 rows of small squares, intended for writing the algorithm.

- 2.6 (4 points)** Écrivez une fonction « *IntToTab* » transformant un entier en un tableau d'entiers. Chaque case contiendra un chiffre. On admettra que le tableau où vous devrez écrire est déjà alloué, et sa taille est également donnée en paramètre.

Le nombre 3204 doit devenir le tableau suivant

3	2	0	4
---	---	---	---

Algorithme :



**SUJET B**  
**ALGORITHMIQUE 1**