

Rattrapage 2022-2023 - CYBER1 (1h30)**Algo et Structure de Données 1**

NOM :

PRÉNOM :

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

- I) Lisez le sujet en entier avec attention
- II) Répondez sur le sujet
- III) Ne détachez pas les agrafes du sujet
- IV) Écrivez lisiblement vos réponses (si nécessaire en majuscules)
- V) Vous devez écrire dans le langage algorithmique classique ou en C (donc pas de Python ou autre)
- VI) Ne trichez pas

1 Questions (5 points)

- 1.1 (1 point) Écrivez l'état d'une file après avoir effectué ces opérations (la file est considérée comme initialement vide), puis, indiquez quel élément sortira de la file lors du prochain « dequeue », ainsi que celui qui sortira en dernier :**

enfiler 1, enfiler 3, défiler, enfiler 4, enfiler 5, défiler, défiler, enfiler 6, enfiler 2, défiler, enfiler 1

6	2	1			
Head		Tail			

Prochain élément qui sortira : 6

Dernier élément qui sortira : 1

Vaguement toléré/Quelques points en moins si et seulement si le prochain et dernier élément qui sortiront sont justes (head = prochain élément qui doit sortir, donc ce n'est pas bon)

1	2	6			
Head					

- 1.2 (1 point) Écrivez l'état d'une pile après avoir effectué ces opérations (la pile est considérée comme initialement vide), puis, indiquez quel élément sortira de la pile lors du prochain « pop », ainsi que celui qui sortira en dernier :**

empiler 1, empiler 3, dépiler, empiler 4, empiler 5, dépiler, dépiler, empiler 6, empiler 2, dépiler, empiler 1

1	6	1			
Head					

Prochain élément qui sortira : 1

Dernier élément qui sortira : 1

1.3 (3 points) En admettant que l'on dispose d'une pile et que l'on insère les données « 1 2 3 4 5 6 » dans cet ordre exclusivement, décrivez les scénarios permettant d'obtenir les sorties suivantes :

exemple : pour « A B C » en entrée, on peut obtenir « B C A » en sortie en faisant :

« push A », « push B », « pop », « push C », « pop », « pop »

On a bien inséré A, puis B, puis C, mais l'ordre de sortie est différent suivant les « pop »

1, 2, 3, 6, 5, 4

push 1, pop, push 2, pop, push 3, pop, push 4, push 5, push 6, pop, pop, pop

3, 2, 4, 5, 1, 6

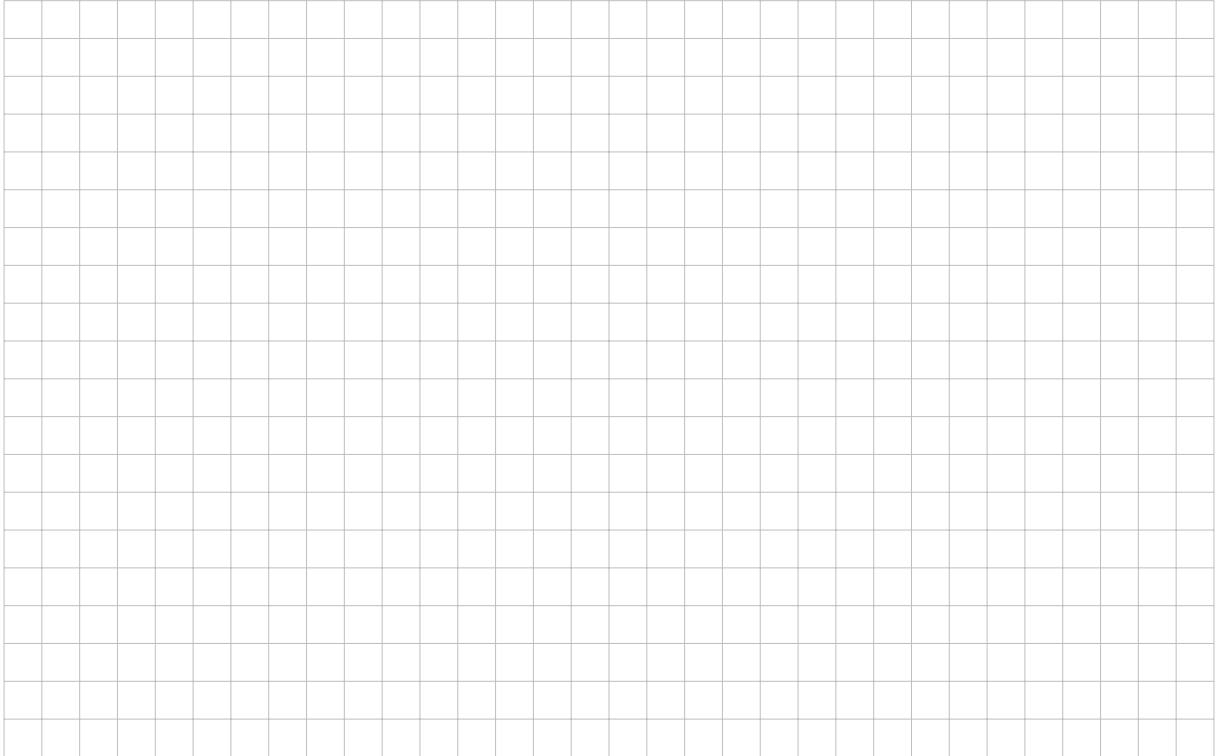
push 1, push 2, push 3, pop, pop, push 4, pop, push 5, pop, pop, push 6, pop

2, 3, 5, 4, 6, 1

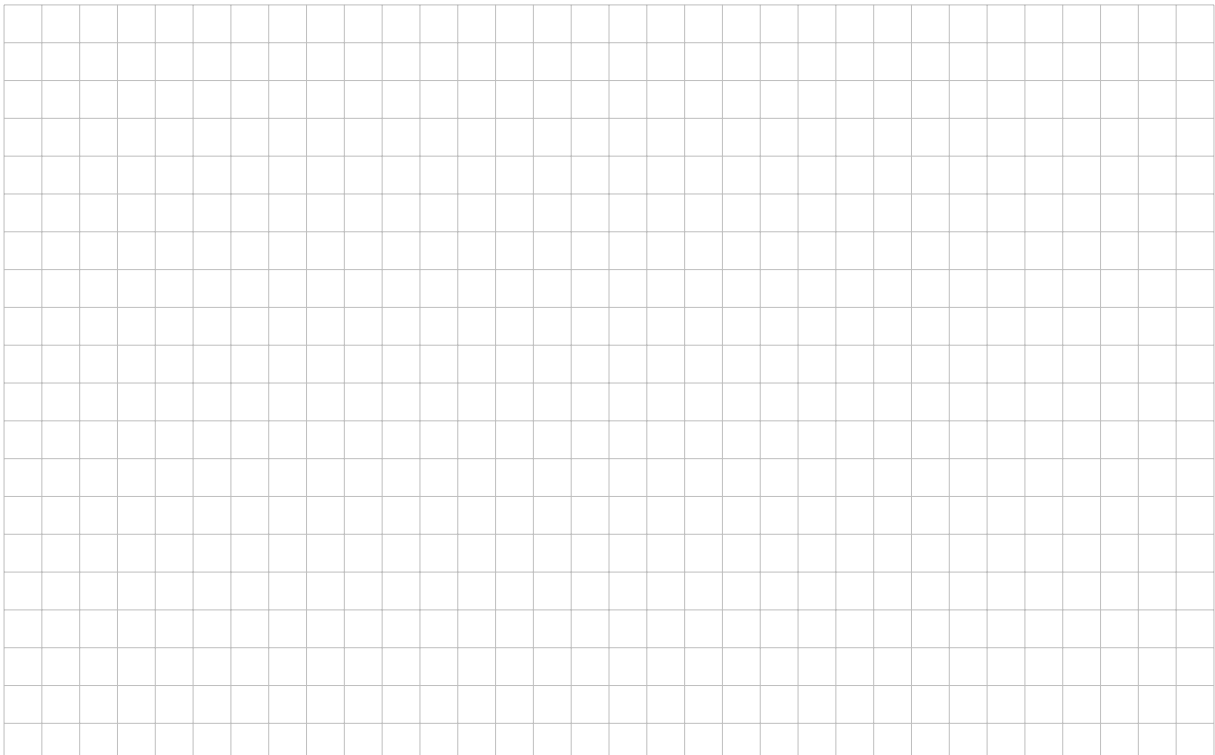
push 1, push 2, pop, push 3, pop, push 4, push 5, pop, pop, push 6, pop, pop

2 Algorithmes (15 points)

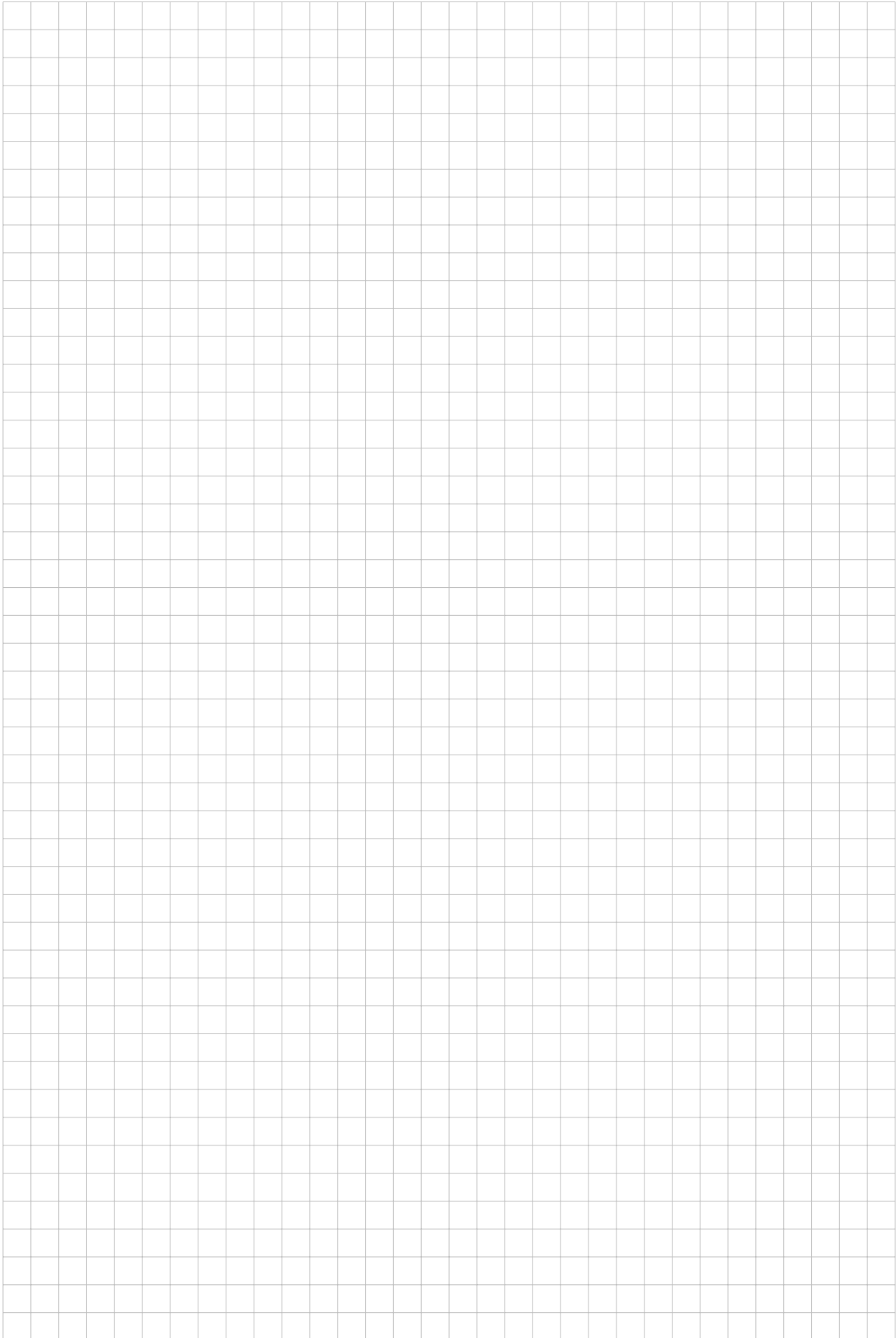
- 2.1 (1,5 point) Écrivez une structure de données « *my_stack_p* » pouvant servir de pile (à base de pointeurs ou de tableaux)



- 2.2 (1,5 point) Écrivez une structure de données « *my_queue_t* » pouvant servir de file (à base de pointeurs ou de tableaux)



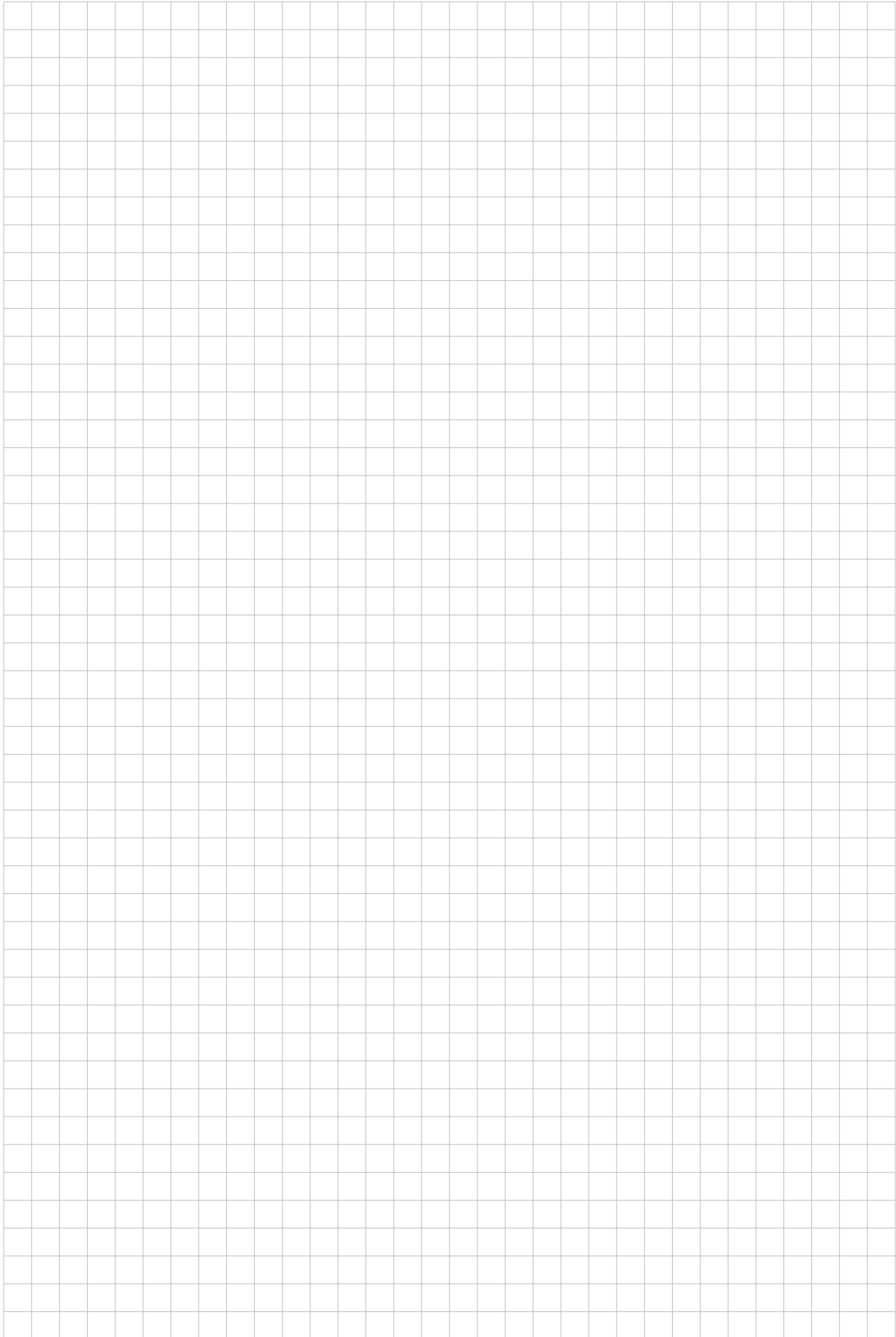
2.3 (3 points) Écrivez une fonction « *push* » pouvant servir à empiler un élément dans votre précédente structure « *my_stack_p* »

A large grid of graph paper, consisting of 20 columns and 30 rows of small squares, intended for writing the code for the 'push' function.

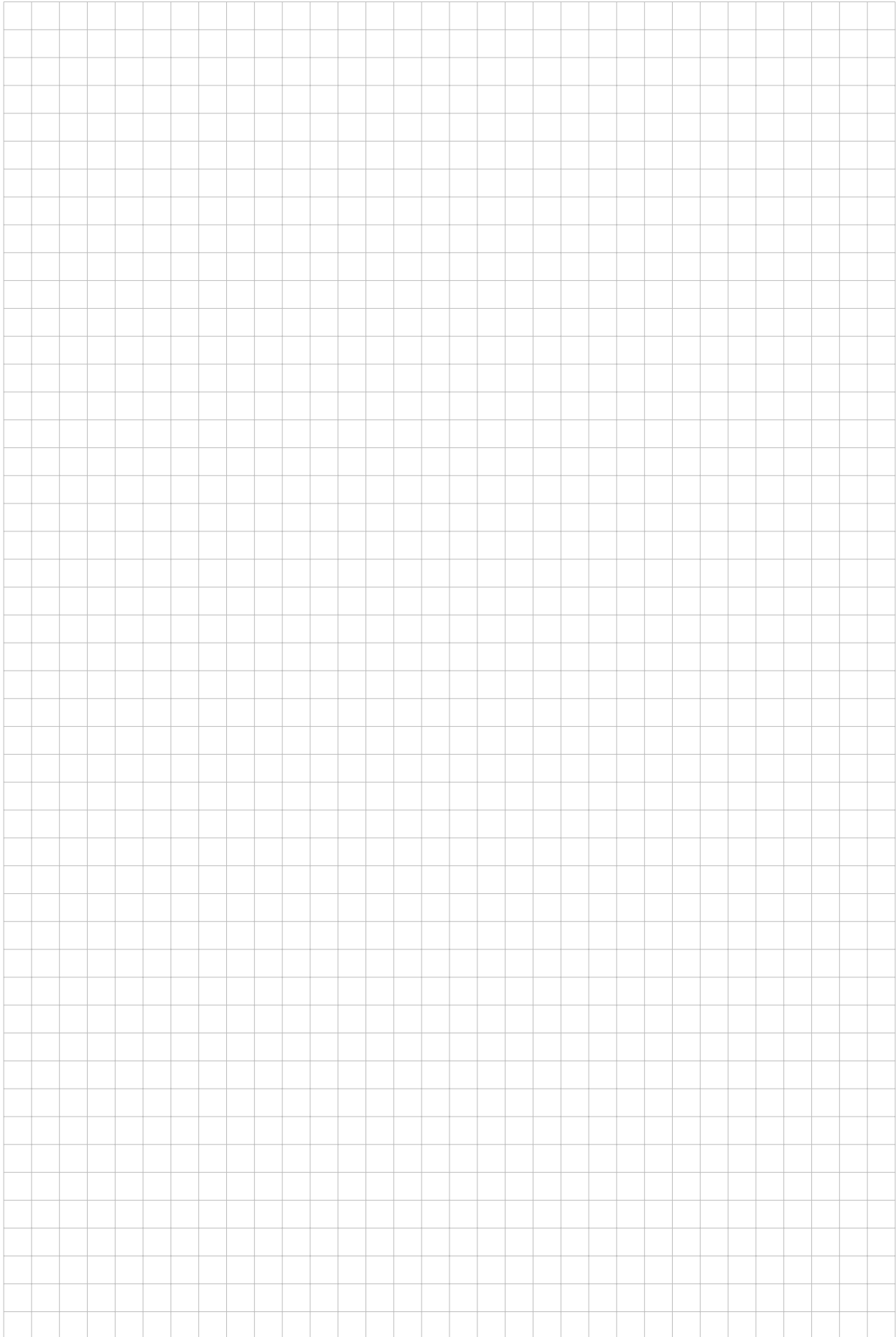
2.4 (3 points) Écrivez une fonction « *pop* » pouvant servir à dépiler un élément dans votre précédente structure « *my_stack_p* »

A large grid of graph paper, consisting of 20 columns and 30 rows of small squares, intended for writing the code for the 'pop' function.

2.5 (3 points) Écrivez une fonction « *enqueue* » pouvant servir à enfiler un élément dans votre précédente structure « *my_queue_t* »

A large grid of graph paper, consisting of 20 columns and 30 rows of small squares, intended for writing the implementation of the enqueue function.

2.6 (3 points) Écrivez une fonction « *dequeue* » pouvant servir à défiler un élément dans votre précédente structure « *my_queue_t* »

A large grid of graph paper, consisting of 20 columns and 30 rows of small squares, intended for writing the implementation of the dequeue function.

RATTRAPAGE ALGORITHMIQUE ET STRUCTURES DE DONNÉES 1