## Examen 2021-2022 - CYBER1 (1h30)

| $\mathbf{A}$ | lgorit | thm | ique | П |
|--------------|--------|-----|------|---|
|              |        |     |      |   |

NOM: PRÉNOM:

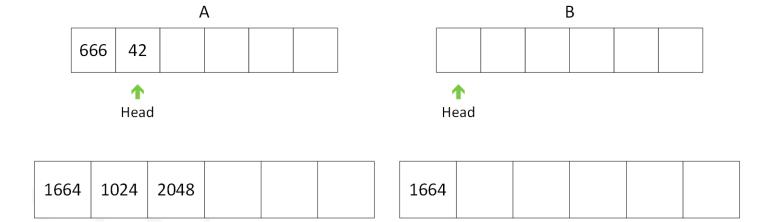
Vous devez respecter les consignes suivantes, sous peine de 0 :

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

## 1 Questions (5 points)

1.1 (1 points) Écrivez l'état des deux files après avoir effectué ces opérations dans cet ordre (n'oubliez pas le(s) pointeur(s) de tête (et de queue) ) :

enfiler 1337, enfiler 2048, défiler, défiler, enfiler 1024, défiler, enfiler 1664



1.2 (0,5 point) Quel élément sortira lors du prochain « dequeue » sur chaque file?

A: 2048 B: 1664

1.3 (0,5 point) Quel élément sortira en dernier de chaque file?

A: 1664 B: 1664

1.4 (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 »
```

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

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

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

- 2 Algorithmes (15 points)
- 2.1 (3 points) Écrivez une structure de données « my\_queue » pouvant servir de file (la structure ne doit pas être statique)

[La structure minimaliste en liste chaînée est tolérée, tout comme les structures plus complexes sous forme de conteneur avec des attributs]

```
struct my_queue_1 {    // Structure minimaliste
  int/void* elt,
 my_queue_1 *next
};
                    // Conteneur lise chainee
struct my_queue_2 {
  int
          nb_elt,
  [my_queue_1 *head,]
  [my_queue_1 *tail,]
 my_queue_1 *queue
};
struct my_queue_3 {    // Conteneur tableau
  int
        nb_elt,
  int
        max_len,
  [int head,]
  [int tail,]
  int
        *array
};
```

2.2 (6 points) Écrivez une fonction « enqueue » pouvant servir à enfiler un élément dans votre précédente structure «  $my\_queue$  »

2 cas possibles : tableau OU liste chaînée

## Vérifier si :

- pointeur NULL est géré en paramètre,
- file **pleine** est gérée en paramètre,
- et le cas normal (**malloc**, et suppression en tête/queue avec réorganisation des éléments suivant).

2.3 (6 points) Écrivez une fonction « dequeue » pouvant servir à défiler un élément dans votre précédente structure «  $my\_queue$  »

2 cas possibles : tableau OU liste chaînée

## Vérifier si :

- pointeur NULL est géré en paramètre,
- file **vide** est gérée en paramètre,
- et le cas normal (free, et suppression en tête/queue avec réorganisation des éléments suivant).