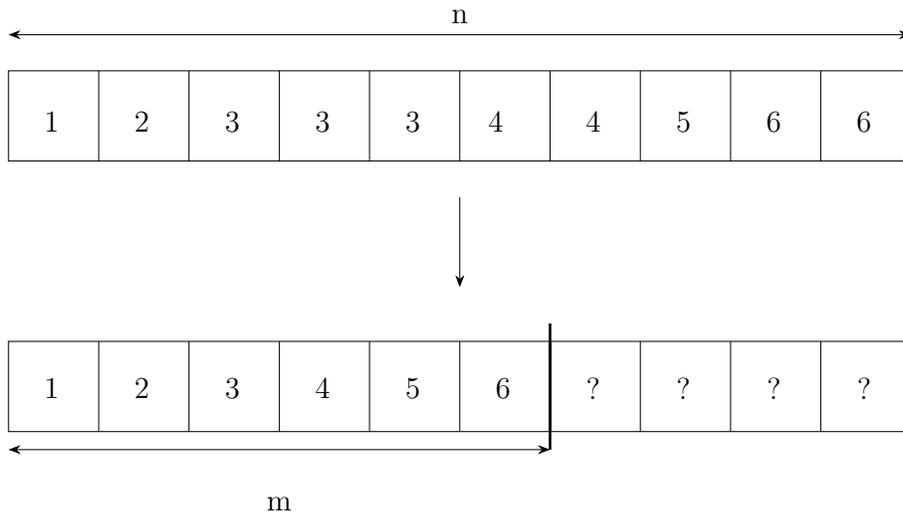


Cours d'introduction à l'informatique

Préparation à l'examen

- (a) Écrire en langage C une procédure prenant en argument une chaîne de caractères et la remplaçant par la chaîne de caractères inversée. Par exemple, pour la chaîne de caractères "Informatique", la procédure la remplacera par "euqitamrofnI".
(b) Calculer la complexité en temps de cette fonction.
- (a) Écrire en langage C une fonction prenant en argument un tableau d'entiers triés, ainsi que sa taille strictement positive. La fonction doit supprimer tous les doublons dans le tableau, de sorte à ce que les doublons ne soient plus que présents une fois et retourner la nouvelle taille du tableau, ne contenant que des entiers différents. La figure suivante montre un exemple de cette opération, où n correspond à la taille du tableau, m correspond à la taille du tableau n'ayant que des éléments différents, et $?$ indique qu'on ne fait pas attention à la valeur de cet élément :



- (b) Par la méthode des invariants, démontrer que l'opération effectuée par la fonction obtenue au point (a) est correcte. Démontrer également

que cette fonction se termine.

3. (a) Expliquer le plus simplement possible ce que calcule la fonction suivante :

```
int f(int n) {
    if (n == 0) {
        return 0;
    } else {
        return n % 10 + f(n / 10);
    }
}
```

- (b) Calculer la complexité en espace de cette fonction.
(c) Écrire une fonction réalisant exactement la même opération, mais sans effectuer d'appel récursif.
4. (a) Écrire en langage C une fraction de code définissant un type structuré capable de représenter une pile d'entiers positifs. Une instance de ce type est composée d'un vecteur d'entiers positifs représentant les éléments qui sont actuellement stockés ainsi que du nombre d'éléments actuellement stockés. Le vecteur d'éléments a une taille maximale de 100.
(b) Écrire une fonction prenant en argument un pointeur vers une instance du type structuré et un entier positif et qui ajoute l'entier au sommet de la pile puis retourne 1 si l'opération s'est correctement déroulée et 0 sinon.
(c) Écrire une fonction prenant en argument un pointeur vers une instance du type structuré et qui enlève l'élément au sommet de la pile avant de le retourner. S'il n'y a pas d'élément dans la pile, la fonction retourne -1.

Note : Vous pouvez programmer des fonctions ou des types de données supplémentaires si votre solution le nécessite.

5. (a) Écrire en langage C une fraction de code définissant un type structuré capable de représenter une liste doublement chaînée d'entiers. Une instance de ce type est composée d'un entier représentant la valeur de l'élément, un pointeur vers l'élément précédent, et un pointeur vers l'élément suivant. Le premier élément de la liste doublement chaînée aura précédent qui pointera vers NULL, et le dernier élément aura suivant qui pointera vers NULL.

Voici un exemple d'une telle liste avec 3 éléments :



- (b) Écrire une fonction prenant en argument un pointeur vers une instance du type structuré, qu'on assumera être le premier élément de la liste doublement chaînée, et un entier n . La fonction ajoutera à la fin de la liste le nouvel entier n dans un élément nouvellement alloué et retournera l'élément créé.
- (c) Écrire une fonction prenant en argument un pointeur vers une instance du type structuré et qui retourne la somme de sa valeur avec la valeur des éléments suivants dans la liste doublement chaînée.
- (d) Écrire une fonction prenant en argument un pointeur vers une instance du type structuré et qui retourne le produit de sa valeur avec la valeur des éléments précédents dans la liste doublement chaînée.
- (e) Écrire une fonction prenant en argument un pointeur vers une instance du type structuré, correspondant au premier élément de la liste doublement chaînée, et qui libérera l'entièreté de la liste.

Note : Vous pouvez programmer des fonctions ou des types de données supplémentaires si votre solution le nécessite.