

Cours d'introduction à l'informatique

Examen blanc 2025-2026

Les réponses doivent être justifiées. Tous les programmes et fragments de code demandés doivent être rédigés en langage C.

Énoncés

1. (a) Écrire une fonction prenant en arguments un tableau t d'entiers supposés triés par ordre croissant, ainsi que la taille $n > 0$ de ce tableau, et retournant le nombre ayant le plus d'occurrences dans le tableau. Si plusieurs nombres ont le même nombre d'occurrences, c'est le plus petit nombre avec ce nombre d'occurrences qui doit être retourné. Par exemple, si le tableau contient $[5, 8, 8, 9, 10]$, alors la fonction doit retourner 8. Si le tableau contient $[2, 2, 5, 5, 8]$, alors la fonction doit retourner 2.
(b) Par la méthode des invariants, démontrer que cette fonction retourne une valeur correcte.
2. En arithmétique, un nombre entier $n > 1$ est dit *déficient* si la somme de ses diviseurs stricts (autre que n) est inférieur à n .
Par exemple, le nombre 17 est déficient, car son seul diviseur est 1. 8 est déficient car la somme de ses diviseurs est $1 + 2 + 4 = 7$. 24 ne l'est pas car la somme de ses diviseurs est $1 + 2 + 3 + 4 + 6 + 8 + 12 = 36$.
(a) Écrire une fonction `deficient` prenant en argument un entier $n > 1$, retournant 1 si n est un nombre déficient et 0 sinon. On demande que la complexité de cette fonction soit particulièrement efficace et doit donc avoir une complexité en temps théorique plus petite que la complexité linéaire par rapport au nombre n .
(b) Calculer la complexité en temps de la fonction obtenue au point (a).
3. (a) Décrire le plus simplement possible l'opération effectuée par la fonction suivante :

```
unsigned f (unsigned v)
{
    if (!v)
        return 0;

    unsigned n = v%10;
    return n + f(v/10);
}
```


(b) Calculer les complexités en temps et en espace de cette fonction.
(c) Écrire une fonction réalisant exactement la même opération, mais sans effectuer d'appel récursif.
4. Une recette est une description structurée permettant de réaliser un plat. Elle est composée d'un ensemble d'étapes ordonnées, décrivant chacune une action spécifique à effectuer.
(a) Définir un type structuré représentant une étape. Ce type structuré doit comprendre
— un pointeur vers une chaîne de caractères donnant les instructions de l'étape,
— un entier donnant le numéro de l'étape,
— un pointeur vers l'étape suivante. (Si aucune étape ne suit, le pointeur est vide.)
(b) Écrire une fonction prenant en arguments un tableau `instructions` de chaînes de caractères constituant les différentes étapes de la recette dans l'ordre, ainsi que `n`, la taille du tableau. On demande que la fonction crée une recette pour les `n` étapes données et retourne un pointeur vers la première étape de cette recette.

- (c) Écrire une procédure prenant en argument un pointeur vers la première étape créée par la fonction du point (b). Cette procédure doit libérer l'ensemble de la mémoire allouée.

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