

Cours d'introduction à l'informatique

Examen d'août 2025

Livres fermés. Durée : 3 h 30

Veillez répondre aux questions sur des feuilles séparées sur lesquelles figurent votre nom, prénom et matricule. Les réponses doivent être justifiées. Tous les programmes et fragments de code demandés doivent être rédigés en langage C.

1. (a) Écrire une fonction prenant en arguments un tableau d'entiers signés t ainsi que sa taille strictement positive n , et qui retourne l'écart entre les deux valeurs extrêmes du tableau. Par exemple, si le tableau contient $[2, -2, 10, 5, 10]$, alors la fonction doit retourner 12. Si le tableau contient $[5, 5]$ ou $[3]$, elle doit retourner 0.
(b) Par la méthode des invariants, démontrer que la valeur calculée par cette fonction est correcte.

2. (a) Écrire une procédure qui inverse l'ordre des caractères dans une chaîne qu'elle reçoit en argument. Par exemple, si on fournit à cette procédure la chaîne de caractères "Informatique", celle-ci doit être modifiée en "euqitamrofnI".
Note importante : Pour résoudre ce problème, il n'est pas permis d'utiliser des fonctions issues de la bibliothèque standard. Vous pouvez en revanche définir des fonctions supplémentaires de votre choix.
(b) Calculer la complexité en temps de la procédure obtenue au point (a).

3. (a) Décrire le plus simplement possible (une phrase suffit) ce que calcule la fonction C suivante :

```
int f(int *a, unsigned b)
{
    if (b <= 1)
        return 1;

    return (a[b - 2] <= a[b - 1]) && f(a, b - 1);
}
```

- (b) Quelle est la complexité en espace de la fonction du point (a) ?
- (c) Écrire une fonction `C` réalisant exactement la même opération que la fonction du point (a), mais sans effectuer d'appel récursif.
4. Un navigateur WWW enregistre l'historique d'une session en mémorisant les pages Web qui ont été visitées, ainsi que le moment où elles ont été visitées. Chaque page Web est identifiée par une chaîne de caractères appelée URL (*Uniform Resource Locator*), par exemple "`https://www.uliege.be`". La date et l'heure d'une visite sont représentées par un seul nombre entier de type `unsigned long` correspondant au nombre de secondes écoulées depuis une date de référence fixée au premier janvier 1970 à 0h00. De plus, pour chaque page faisant partie de l'historique, on conserve un moyen d'accès à la page précédente et à la page suivante (par rapport à l'ordre chronologique où elles ont été visitées), de façon à pouvoir naviguer en arrière ou en avant.
- (a) Écrire un fragment de code définissant un type structuré capable de représenter une page Web contenue dans un historique. Ce type structuré doit comprendre
- un pointeur vers une chaîne de caractères donnant l'URL de la page,
 - la représentation de la date et l'heure de visite de la page,
 - deux pointeurs vers la page précédente et la page suivante dans l'historique. (Si l'une de ces pages n'existe pas, le pointeur concerné est alors vide.)
- (b) Écrire une fonction prenant en arguments un tableau `url` de chaînes de caractères constituant les URLs de pages visitées, dans l'ordre chronologique de leur visite, un tableau `temps` d'entiers contenant les dates et heures de visite de ces pages (en d'autres termes, `temps[i]` contient le moment où la page `url[i]` a été visitée), et la taille `n` commune de ces deux tableaux, supposée non nulle. On demande que la fonction crée un nouvel historique de navigation pour les `n` pages fournies, en allouant dynamiquement la mémoire nécessaire, et retourne un pointeur vers la représentation de la première page faisant partie de cet historique.
- Note* : Vous pouvez programmer des fonctions ou des types de données supplémentaires si votre solution le nécessite, et utiliser des fonctions issues de la bibliothèque standard.
- (c) Écrire une fonction prenant en argument un pointeur vers n'importe quelle page d'un historique de navigation, et qui retourne le nombre de secondes écoulées entre les moments où la première et la dernière page de cet historique ont été visitées.