

Introduction à la programmation  
TP2 - Les tableaux et les structures  
Partie pratique (25%)

420-BC6-AG

Date de remise : 3 octobre 2018

*Le but de ce TP est de mettre en pratique la manipulation de tableaux et de structures.*

À faire seul.

Remettez les fichiers suivants dans un dossier compressé sur LÉA :

- `parenthesage.c`
- `magique.c`
- `demain.c`
- Votre diagramme de séquence pour la question 3.

Dans les commentaires de chaque fichier de code source, donnez trois cas d'entrée que vous avez testé et le résultat de votre programme pour ces entrées.

Vous devez respecter les règles et les conventions de nomenclature, d'indentation, de gestion des nombres magiques et de documentation.

Le plagiat, partiel ou total, entraînera la note de 0.

# 1 Parenthésage

Programmez une fonction qui reçoit une chaîne de caractères et qui retourne vrai si elle est correctement parenthésée et faux sinon.

Une expression est bien parenthésée si elle contient autant de parenthèses fermantes qu'ouvrantes et que chaque parenthèse fermante vient après une parenthèse ouvrante correspondante.

La signature de la fonction est la suivante :

---

```
int estBienParenthesee(char expression[]);
```

---

Voici des exemples d'expressions bien parenthésées :

---

```
"((((alpha))))bravo"  
"charlie () delta"  
"echo foxtrot"  
"((golf) (hotel, juliette)) kilo ( lima, mike)"
```

---

Voici des exemples d'expressions mal parenthésées :

---

```
")!("  
"(((november"  
"oscar :)"
```

---

## 2 Le carré presque magique

Un carré presque magique est une matrice carrée  $n \times n$  dont la somme des rangées, colonnes et diagonales est égale. Programmez une fonction qui retourne vrai si la matrice reçue est un carré presque magique et faux sinon.

Voici quelques exemples de carrés presque magiques :

$$\begin{pmatrix} 16 & 3 & 2 & 13 \\ 5 & 10 & 11 & 8 \\ 9 & 6 & 7 & 12 \\ 4 & 15 & 14 & 1 \end{pmatrix} \quad \begin{pmatrix} 8 & 8 & 8 \\ 8 & 8 & 8 \\ 8 & 8 & 8 \end{pmatrix} \quad \begin{pmatrix} 4 & 9 & 2 \\ 3 & 5 & 7 \\ 8 & 1 & 6 \end{pmatrix}$$

---

```
// R est le nombre de rangees, C est le nombre de colonnes.
int estPresqueMagique(const int R, const int C, int
    matrice[R][C])
{ /* Votre code ici */ }

int main()
{
    int matrice[3][3] = {{2,7,6},{9,5,1},{4,3,8}};

    if (estPresqueMagique(3, 3, matrice))
    {
        printf("La matrice est presque magique.\n");
    }
    else
    {
        printf("La matrice n'est pas presque magique.\n");
    }
}
```

---

### 3 Demain

Rédigez les fonctions `bisextile`, `nbJoursDansMois`, `estValide` et `demain` sur les dates. La fonction `demain` doit faire appel aux trois autres (directement ou indirectement). Fournissez le diagramme de séquence pour la fonction `demain`.

---

```
#include <stdio.h>

typedef struct Date Date;
struct Date{
    int jour;
    int mois;
    int annee;
};

int bisextile(int annee)
{ /* Votre code ici */ }

int nbJoursDansMois(int mois, int annee)
{ /* Votre code ici */ }

int estValide(Date date)
{ /* Votre code ici */ }

Date demain(Date aujourd'hui)
{ /* Votre code ici */ }

int main()
{
    Date date;
    printf("Entrez une date (jj/mm/aaaa)\n");
    scanf("%d/%d/%d", &date.jour, &date.mois, &date.annee);

    Date lendemain = demain(date);
    printf("Lendemain : %d/%d/%d\n", lendemain.jour,
        lendemain.mois, lendemain.annee);
}
```

---

## 4 Bonus [0, 5 ou 10 points]

### 4.1 Parenthèses, crochets et accolades

Adaptez la fonction `estBienParenthesee` pour qu'elle valide qu'en plus d'avoir un parenthésage correct, l'expression a des crochets et des accolades correctes. Attention, lorsqu'un bloc externe se ferme, tous les blocs internes qui ont été ouverts doivent avoir été fermés.

Voici des exemples d'expressions bien parenthésées :

---

`"[()]", "{[()]}"`

---

Voici des exemples d'expressions mal parenthésées :

---

`"[(])", "{[({})]}"`

---

### 4.2 Carré vraiment magique

Un carré vraiment magique est une matrice carrée  $n \times n$  dont la somme des rangées, colonnes et diagonales est égale. En plus, chaque nombre de 1 à  $n^2$  doit apparaître une et une seule fois dans la matrice. Programmez une fonction qui retourne vrai si la matrice reçue est un carré magique et faux sinon.

## Critères d'évaluation

- [15 pts] Application des règles de syntaxe et de sémantique ainsi que des conventions de lisibilité et de qualité propres au C.
- [12 pts] Adaptation et traduction correcte de l'algorithme en C.
- [12 pts] Modélisation correcte du traitement.
- [6 pts] Modélisation correcte des données.
- [6 pts] Documentation appropriée.
- [3 pts] Éclatement approprié en fonctions.
- [3 pts] Banque de tests appropriée.
- [3 pts] Compilation correcte et sans erreurs ni avertissements.

Pour un total de 60 points qui seront ramenés sur 25.

## **Politique de retard**

À partir de la PIEA, pour l'enseignement aux adultes, la politique de retard dans les remises de travaux a été précisée de la manière suivante :

2.3.3.1 L'étudiant-e qui remet une production écrite (rapport de laboratoire, analyse, dissertation, recherche, etc.) ou une production concrète (vidéo, œuvre d'art, montage, etc.) en retard est pénalisé-e.

2.3.3.2 Pour les travaux dont le délai de production est d'une semaine (7 jours) ou moins, le professeur peut refuser le travail et inscrire la note zéro.

2.3.3.3 Pour les travaux dont le délai de production est de plus d'une semaine (8 jours et plus), le professeur peut enlever jusqu'à 10 % des points par jour de retard, incluant les jours de fin de semaine.

2.3.3.4 Aucun travail ne sera accepté après la remise des travaux corrigés.

## **Politique de plagiat**

À partir de la PIEA, pour l'enseignement aux adultes, la politique de plagiat dans les remises de travaux a été précisée de la manière suivante :

### **3.3 Le plagiat (Dionne, 2013, p.199)**

Le plagiat est l'acte de faire passer pour siens des textes, des contenus, des réponses ou des idées d'autrui, sans citer la source. Par exemple :

3.3.1 Copier le travail d'une autre personne en totalité ou en partie.

3.3.2 Utiliser l'œuvre d'autrui, des passages ou des idées de celle-ci sans en citer la source.

3.3.3 Copier une page (ou un segment de page) sur le Web sans en mentionner la source.

3.3.4 Résumer l'idée originale d'un auteur en l'exprimant dans ses propres mots, mais en omettant d'en indiquer la source.

3.3.5 Traduire partiellement ou totalement un texte sans en mentionner la provenance.

3.3.6 Utiliser le travail d'une autre personne et le présenter comme le sien, et ce, même si cette personne a donné son accord.

3.3.7 Paraphraser un texte sans mentionner la source.

### **3.4 La fraude, la tricherie et la tentative de tricherie**

La fraude est un acte de tromperie qui vise l'obtention d'un avantage personnel, parfois au détriment des autres. Par exemple :

3.4.1 Obtenir les questions ou les réponses d'un examen avant l'évaluation.

- 3.4.2 Posséder du matériel non autorisé (calculatrice, formulaire, notes).
- 3.4.3 Obtenir une aide quelconque inappropriée ou non autorisée.
- 3.4.4 Utiliser du matériel non autorisé.
- 3.4.5 Consulter la copie d'un·e autre étudiant·e qu'il·elle soit complice ou non.
- 3.4.6 Réutiliser un travail produit dans un autre cours sans avoir obtenu au préalable l'accord du professeur.
- 3.4.7 Inventer des données dans le cadre d'un travail.
- 3.4.8 Remplacer un·e étudiant·e lors d'un examen.
- 3.4.9 Se faire remplacer par une autre personne lors de l'examen.
- 3.4.10 Acheter un travail et le déposer comme étant le sien.

### 3.5 Conséquences

Tout plagiat, toute fraude, toute tentative de plagiat ou de fraude, toute coopération à un plagiat ou à une fraude et toute présence de matériel non autorisé entraînent la note zéro pour l'activité concernée et peuvent entraîner une réévaluation des résultats antérieurs. S'il y a récurrence, le cas est automatiquement signalé au comité d'admission; l'étudiant·e pourrait alors se voir exclu·e du Collège et ce dernier pourrait ne pas recommander la sanction de cet·te étudiant·e. (La définition de plagiat et tricherie est donnée à la partie 3.3 de la PIEA)

### **Politique linguistique**

Le travail doit être produit en langue française (commentaires et documentation).

### **Politique de contenu**

Sont à proscrire tous les contenus à caractère irrespectueux, diffamatoire ou explicite, qu'ils soient sexuels, racistes, religieux ou autres. Tout contenu devra être validé par le professeur avant d'être diffusé.