

DM2 Info : consignes

À propos de ce devoir

Fichiers

Ce devoir est à déposer sous forme d'archive sur le site **Cahier de prépa** avant le mardi 10 octobre 23h30. Aucun rendu par mail n'est accepté.

Dans le menu **Informatique** de **Cahier de prépa**, cliquer sur **Transfert de document**. Une page titrée DM1 apparaît.

Il y a un seul fichier à glisser dans l'archive, il s'agit de **dm2.c**. Si une question demande autre chose que l'écriture d'un code, mettre la rédaction de sa réponse en commentaire.

Les fichiers sont inclus dans une archive **.zip** respectant la règle de nomenclature décrite suivante :

L'élève Pierre Dupont doit constituer une archive **pierre_dupont.zip** contenant ses deux fichiers. On autorise les réponses par binôme. Si Pierre Dupont et Marie-Chloé Léonidas de la Tour s'entendent bien, ils renvoient un fichier **marie_chloe_leonidas_de_la_tour_ET_pierre_dupont.zip**. Pas d'accent ni de trait d'union.

Pour compresser au format **.zip** le répertoire **Acompresser** et son contenu dans l'archive **pierre_dupont.zip**, se placer dans le répertoire parent et entrer :

```
zip pierre_dupont.zip Acompresser/*
```

Nom(s), email(s), compilation

En commentaire, au début de tous vos fichiers source, vous écrivez les noms des auteurs séparés par des virgules et les emails de chacun (voir figures 1 et 2). **Cette règle n'a pas été assez respectée lors du dernier devoir.**

En commentaire, au début de votre fichier source **.c**, vous écrivez la commande de compilation du fichier.

Un fichier qui ne passe pas l'épreuve de la compilation avec **-Wall -Werror=vla** est noté E. **Dans le précédent devoir, la manipulation de tableaux multidimensionnels statiques rendait impossible l'usage de l'option sans VLA. Ce n'est pas le cas pour le devoir 2 : tous les tableaux sont dynamiques.** Les Warning font baisser la note.

Pas de saisie

Aucun **scanf** dans le code rendu.

```

1 // Nom : Pierre Dupont
2 // email : pierre.dupont@gmail.com
3 // gcc dml.c -Wall -Werror=vla -o exo1
4

```

FIGURE 1 – Pierre a fait son devoir seul

```

1 /*
2 Nom : Marie–Chloé Léonidas de la Tour; Pierre Dupont
3 email : pierre.dupont@gmail.com; marie.ldelatour@hotmail.fr
4 compilation : gcc dml.c -Wall -Werror=vla -o exo1
5 */
6
7

```

FIGURE 2 – Pierre et Marie ont travaillé ensemble

Consignes d’écriture de code

Il faut commenter son code. Hors fonctions de tests, les fonctions dont le code est long (plus d’une dizaine de lignes) doivent être décomposées en fonctions auxiliaires pour faciliter la lecture du travail rendu.

Un code lisible est un code commenté : la raison de telle instruction conditionnelle ou telle boucle est expliquée. On évite de paraphraser le code. Les variables locales ont des noms courts qui expliquent leur signification (`n` pour un entier, `cpt` pour un compteur etc.).

Un code long doit être factorisé en sous-fonctions aux noms explicites. **Pour le devoir 2, les fonctions `resize`, `strassen` et sans doute `merge` nécessitent plusieurs fonctions auxiliaires.** Une bonne habitude est de préfixer les noms de fonctions auxiliaires par un underscore (touche 8) :

```

1 void _aux1(int i, int j, int **t){
2     // some code
3 }
4
5 float _aux2(int ***t, int n){
6     // some code
7 }
8
9 float myfunction(int n, int **t){
10    // ...
11    int i=0, j=1;
12    _aux(i, j, t);
13    float x = _aux2(j, &t);
14    // etc.
15 }

```

Tests et main

Pour chaque fonction demandée en C, il faut introduire une procédure de tests unitaires correspondante. **Cependant, pour le devoir 2, les tests sont fournis par le professeur.**

Le code ne fait pas appel à la fonction `scanf` et le `main` est presque vide : les tests sont effectués par des fonctions dédiés.

Note

Les devoirs maisons sont toujours corrigés et une note indicative (A,B,C,D,E) est fournie. Ces notes ne sont pas prises en compte dans le bulletin. Il ne faut donc pas s'alarmer si votre performance n'est pas bonne.

Il est totalement improductif de recopier la réponse d'un camarade plus à l'aise : vous n'apprendrez rien de la sorte et vous me ferez perdre mon temps. Si vous n'arrivez pas à tout faire, ne faites pas tout. Rendez un travail personnel (on apprend en réfléchissant, pas en devenant expert du copié-collé) de façon à ce que vos points forts soient soulignés et les points à travailler, identifiés.

Songez qu'il existe des outils de comparaison de fichier. Si une fraude est avérée, elle peut être mentionnée dans le bulletin.