

Nom : _____

Prénom : _____

Numéro étudiant :

--	--	--	--	--	--	--	--

Les fichiers

23 mars 2021

A lire absolument :

1. L'objectif n'est pas d'apprendre la correction par cœur, mais de comprendre les mécanismes mis en œuvre. Cela vous permettra de vous adapter face à un problème nouveau.
2. En particulier, vous devez être capable de refaire l'intégralité du sujet, seul, sans aucune aide ni support.
3. Votre travail sera corrigé automatiquement par l'outil de correction automatique CAT. Cela implique que vous devez respecter scrupuleusement les consignes de chaque exercice. Faites très attention aux messages qu'il vous est demandé d'afficher. Un espace en trop, un saut de ligne en moins et l'exercice risque d'échouer.
4. L'enseignant voit votre activité sur le site, ainsi que l'historique de vos dépôts. Pensez à déposer votre travail régulièrement afin qu'il puisse vous apporter des conseils personnalisés.
5. Si l'enseignant vous demande de rendre votre travail sur papier, vous devez répondre directement sur le sujet en respectant absolument la zone prévue à cet effet. Tout ce qui se trouve en dehors de la zone sera ignoré.
6. Si le sujet contient un QCM, vous devez colorier les cases avec un stylo bleu ou noir. Les autres couleurs seront ignorées.
7. Chaque feuille est identifiée de manière unique. Vous pouvez donc rendre votre sujet avec les feuilles mélangées, mais il est préférable de les trier car cela vous permet de vérifier que vous n'en avez pas oublié une.
8. Si vous faites face à un problème, un bug, une erreur ou que vous souhaitez participer à l'amélioration de la plateforme, envoyez un mail à l'adresse suivante : support-cat@liste.parisnanterre.fr

Ne rien écrire dans cette zone



Dans ce TD, nous allons manipuler des fichiers texte. N'hésitez pas à tenter des choses, à créer des fichiers en liens avec vos projets etc. Bref, soyez curieux.

Ecriture simple dans un fichier.

Ecrire une fonction qui crée un fichier texte nommé « exemple.txt ». Ecrivez dans ce fichier le message suivant : "Vive la MIAGE!"

```
1 void ecriture_simple();
```

Ne rien écrire dans cette zone



Ecriture d'une table de multiplication dans un fichier

Ecrire une fonction qui crée un fichier texte nommé « table.txt ». Ce fichier doit contenir la table de multiplication pour le nombre passé en argument, présentée sous la forme suivante :

```
0 x 5 = 0
1 x 5 = 5
2 x 5 = 10
3 x 5 = 15
4 x 5 = 20
5 x 5 = 25
6 x 5 = 30
7 x 5 = 35
8 x 5 = 40
9 x 5 = 45
10 x 5 = 50
```

```
1 void table(int nb);
```

Ne rien écrire dans cette zone



Affichage du contenu d'un fichier.

Ecrire une fonction qui affiche le contenu d'un fichier texte dont le nom est passé en argument. Pour cela, vous devez utiliser la fonction `fgetc`.

```
1 void affichage_fichier(char * nom);
```

Ne rien écrire dans cette zone



Affichage du contenu d'un fichier et des numéros de ligne.

Ecrire une fonction qui affiche le contenu d'un fichier texte dont le nom est passé en argument en faisant précéder chaque ligne par son numéro suivi de :. Exemple :

```
1:coucou les gens !  
2:  
3:Voici un exemple avec 3 lignes !
```

```
1 void affichage_fichier_lignes(char * nom);
```

Ne rien écrire dans cette zone



Analyse et traitement du contenu d'un fichier.

Ecrire une fonction qui affiche le contenu d'un fichier texte en ignorant les lignes de commentaires et en supprimant les blancs en début de ligne. Les caractères blancs sont les espaces et les tabulations. Les lignes dont le premier caractère non blanc est un dièse (#) seront considérées comme des commentaires. Ainsi, le fichier suivant :

```
# Une famille
  Raymonde
Robert
  # et leurs enfants :
Jules
  Jim
```

Sera affiché comme ceci :

```
Raymonde
Robert
Jules
Jim
```

```
1 void affichage_sans_commentaires(char * nom);
```

Ne rien écrire dans cette zone



Le chiffrement de cesar

Ecrire une fonction qui Implémente le chiffre de César. Pour cela vous devez lui passer le nom du fichier source, le nom du fichier de destination ainsi que la clé de décalage. Il n'est pas nécessaire de demander s'il s'agit d'un chiffrement ou d'un déchiffrement puisque le décalage peut être négatif. Vous allez lire ensuite dans le fichier source, caractère par caractère, et placer à chaque fois dans le fichier de destination le caractère décalé.

```
1 void cesar(char * source, char * destination, int decalage);
```

Ne rien écrire dans cette zone





Ne rien écrire dans cette zone

