

Nom : \_\_\_\_\_

Prénom : \_\_\_\_\_

Numéro étudiant :

|  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|
|  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|

# Un premier programme en langage C

9 septembre 2017

## 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](mailto:support-cat@liste.parisnanterre.fr)

Ne rien écrire dans cette zone



### Objectifs.

Dans ce TD, nous allons manipuler un premier programme afin de vous faire découvrir les bases de la programmation en langage C. Nous allons utiliser l'outil C.A.T qui corrige automatiquement vos exercices. Vous le trouverez ici : <http://savoircoder.fr/cat/>.

- Le premier exercice doit vous faire découvrir l'environnement de programmation. Il n'est pas corrigé automatiquement.
- Pour les autres exercices, vous devez à chaque fois réaliser un programme complet (bibliothèques, main etc ...). Pour cela, vous devez d'abord réaliser l'exercice sur votre ordinateur. Ensuite :
  1. faites un copier-coller dans l'outil C.A.T, dans la zone prévue à cet effet.
  2. Cliquez ensuite sur le bouton vert « Envoyer » . Ce bouton se trouve juste au dessus des exercices.
  3. Sur la partie droite de la page, dans les résultats, vous devriez ensuite voir quels sont les exercices réussis, et ceux qui ne le sont pas.
  4. Pour obtenir des précisions sur vos erreurs, vous pouvez cliquer sur le résultat.

Ne rien écrire dans cette zone



# Exercice 1 : Découverte de l'environnement de programmation



## Un exemple de programme

Nous allons observer à quoi ressemble le code source d'un programme réalisé en langage C et découvrir notre premier environnement de développement.

1. Téléchargez le fichier TD1.c sur la plate-forme, et enregistrez le dans votre dossier de travail.
2. Lancez le logiciel Dev C++ (ou Code::Blocks ou n'importe quel autre IDE) et ouvrez le fichier précédent à l'aide de ce programme. Observez les différentes couleurs. Il s'agit de la coloration syntaxique.
3. Essayez d'ouvrir ce même fichier TD1.c avec un éditeur de texte basique (**bloc-notes** par exemple sous Windows). Que constatez vous ?



## Du code source au programme exécutable

1. Trouvez, dans le menu de votre IDE, comment compiler le programme.
2. Trouvez, dans le menu de votre IDE, comment exécuter le programme compilé.
3. Il existe généralement une option pour uniquement compiler le code source, une autre pour exécuter le programme généré et une troisième permettant de compiler le code source ET exécuter le programme.
4. Observez dans votre dossier de travail la présence d'un fichier exécutable (portant l'extension .exe)

Ne rien écrire dans cette zone



## Exercice 2 : Observation du code source



### Des commentaires pour lire plus facilement un code source

Attention, dans cet exercice vous devez respecter les messages, ne pas mettre d'accents et respecter la casse (c'est-à-dire les minuscules et les majuscules).

1. Placez un commentaire multi-lignes juste avant le programme principal. Votre commentaire devra contenir le message suivant : « programme principal ».
2. Placez un commentaire multi-lignes juste avant les directives du pré-processeur. Votre commentaire devra contenir le message suivant : « directives du pre-processeur ».
3. Placez un commentaire mono-ligne à la fin de chaque ligne réalisant un affichage. Votre commentaire devra contenir le message suivant : « affichage d'un message ».
4. Placez un commentaire mono-ligne à la fin de chaque ligne réalisant une saisie au clavier. Votre commentaire devra contenir le message suivant : « saisie d'une valeur au clavier ».
5. Placez un commentaire mono-ligne à la fin de chaque ligne réalisant une déclaration de variable. Votre commentaire devra contenir le message suivant : « declaration d'une variable ».
6. Placez un commentaire mono-ligne à la fin de chaque ligne réalisant une initialisation de variable. Votre commentaire devra contenir le message suivant : « initialisation de variable ».

Ne rien écrire dans cette zone



## Exercice 3 : Un premier programme



### Affichage d'un message

La fonction `printf` permet d'afficher un message. Ce message doit être encadré par deux guillemets `"`. Exemple :

```
printf("Hello_world!\n");
```

A noter : la présence d'un `\n` dans le message à afficher provoque un retour à la ligne.



### Saisie d'une valeur au clavier

Lors d'une interaction avec l'utilisateur, il est fréquent de lui demander de saisir des données au clavier. Pour saisir un entier au clavier et l'enregistrer dans une variable dont le nom est `x` :

```
scanf("%d",&x);
```

A noter : le `%d` indique que l'on souhaite saisir une variable entière. Pour un nombre à virgule, il faut le remplacer par `%f` et par `%c` pour un caractère.

Vous devez écrire un programme qui demande à l'utilisateur de rentrer un entier au clavier, puis calcule et affiche son carré.

Ne rien écrire dans cette zone



## Exercice 4 : Conversion euros-dollars

Écrire un programme qui convertit les euros en dollars. L'utilisateur donnera au programme la somme en euros qu'il désire convertir puis le taux de conversion. Le programme affichera ensuite la somme correspondante en dollars.

Ne rien écrire dans cette zone



## Exercice 5 : Indice de Masse Corporelle

Écrire un programme qui demande à l'utilisateur sa taille (en m) et son poids (en kg) puis qui affiche à l'écran son indice de masse corporelle (IMC) calculé par la formule suivante :

$$IMC = \frac{poids}{taille^2}$$



### Pour en savoir plus

L'indice de masse corporelle (IMC) est un indicateur de la corpulence d'une personne. Il a été inventé par un scientifique belge, Adolphe Quetelet. [https://fr.wikipedia.org/wiki/Indice\\_de\\_masse\\_corporelle](https://fr.wikipedia.org/wiki/Indice_de_masse_corporelle)

Ne rien écrire dans cette zone



## Exercice 6 : Conversion temporelle

Écrire un programme qui lit un nombre de secondes entier fourni par l'utilisateur et qui affiche cette durée en jours, heures, minutes et secondes au format « x jours, x heures, x minutes et x secondes. »

Par exemple, si l'utilisateur rentre la valeur 200000, le message suivant devra s'afficher :  
« 2 jours, 7 heures, 33 minutes et 20 secondes. »

Ne rien écrire dans cette zone





## Exercice 7 : Affichage d'une somme et d'un produit

Écrire un programme qui demande à l'utilisateur deux nombres compris entre 1 et 99 et qui affiche à l'écran leur somme et leur produit sous la forme suivante : (les espaces sont matérialisés par des `_` pour vous aider).

```
Somme_ _ _ _ _ _ _ _ _ _Produit
_ _ _ _ _11_ _ _ _ _ _ _ _ _11
+_ _ _ _ _13_ _ _ _ _x_ _ _ _ _13
-----_ _ _ _ _-----
_ _ _ _ _24_ _ _ _ _ _ _ _ _143
```

Ne rien écrire dans cette zone



Ne rien écrire dans cette zone

