



Piscine C

Colle 00

Staff 42 [piscine@42.fr](mailto:piscine@42.fr)

*Résumé: Ce document est le sujet de la colle 00 de la piscine C de 42.*

# Table des matières

<b>I</b>	<b>Consignes</b>	<b>2</b>
<b>II</b>	<b>Préambule</b>	<b>4</b>
<b>III</b>	<b>Sujet commun</b>	<b>5</b>
<b>IV</b>	<b>Colle 00</b>	<b>6</b>
<b>V</b>	<b>Colle 01</b>	<b>8</b>
<b>VI</b>	<b>Colle 02</b>	<b>9</b>
<b>VII</b>	<b>Colle 03</b>	<b>10</b>
<b>VIII</b>	<b>Colle 04</b>	<b>11</b>

# Chapitre I

## Consignes

- Chaque membre du groupe peut inscrire le groupe en soutenance.
- Le groupe doit être inscrit en soutenance.
- Toute demande de précision sur un des sujets compliquera le sujet.
- Vous devez suivre la procédure de rendu pour tous vos exercices.
- Le sujet peut changer jusqu'à une heure avant le rendu.
- Les exercices sont très précisément ordonnés du plus simple au plus complexe. En aucun cas nous ne porterons attention ni ne prendrons en compte un exercice complexe si un exercice plus simple n'est pas parfaitement réussi.
- La Moulinette compile avec les flags -Wall -Wextra -Werror, et utilise gcc.
- Si votre programme ne compile pas, vous aurez 0.
- Les exercices de colle sont à réaliser par groupe de 2, 3 ou 4.
- Vous trouverez dans les fichiers group\_promo.txt la liste des groupes imposés avec le sujet.
- Vous devrez donc réaliser le sujet indiqué avec les binômes imposés et vous présenter en soutenance à l'heure dite avec tous vos binômes.
- Lors de la soutenance, le projet devra être terminé. Les soutenances servent à présenter et à expliquer votre travail dans les moindres détails.
- Chaque membre du groupe devra parfaitement être au courant du travail réalisé, chacun des membres sera interrogé, la note du groupe étant basée sur les moins bonnes explications.
- Évidemment, vous devez tout faire pour prendre contact avec vos binômes : téléphone, mail, pigeon voyageur, séance de spiritisme, etc. Aucune excuse ne sera acceptée en ce qui concerne les problèmes de groupe.
- Si après avoir vraiment tout essayé un de vos binômes reste injoignable : réalisez votre colle on s'arrangera en soutenance. Même si c'est le chef de groupe : vous avez tous accès au dépôt.

- Vous pouvez, à titre optionnel, réaliser plusieurs sujets pour avoir un éventuel bonus.
- La Moulinette n'est pas très ouverte d'esprit. Elle ne cherche pas à comprendre le code qui ne respecte pas la Norme. La Moulinette utilise le programme `norminette` pour vérifier la norme de vos fichiers. Comprendre par là qu'il est stupide de rendre un code qui ne passe pas la `norminette`.



Il faudra absolument avoir le sujet obligatoire réalisé parfaitement pour prétendre aux sujets bonus.



Pour cette journée, la `norminette` doit être lancée avec le flag  
-- `CheckForbiddenSourceHeader`. La moulinette l'utilisera aussi.

# Chapitre II

## Préambule

Voici les paroles du générique de Minus et Cortex :

Minus : Dis Cortex, tu veux faire quoi cette nuit?

Cortex : La même chose que chaque nuit, Minus: tenter de conquérir le monde!

C'est Minus et Cortex

C'est Cortex et Minus

L'une est plein d'astuce

L'autre un vrai nimbus

Deux souris diaboliques

Du génie génétique

Quelles canailles,

Ces p'tites souris cobayes, -bayes, -bayes, -bayes,

-bayes, -bayes, -bayes, -bayes,

-bayes!

Dans leur tête elles projettent

Des plans sur la comète

Pour partir à la conquête

De toute la planète

C'est Cortex et Minus

C'est Minus et Cortex

Qui ont le réflexe

De vouloir sans complexes

Tendre une souricière

À la terre toute entière

Quelles canailles,


Ces p'tites souris cobayes, -bayes, -bayes, -bayes,

-bayes, -bayes, -bayes, -bayes.

Plutôt que de conquérir le monde, vous allez vous employer à conquérir cette colle!

# Chapitre III

## Sujet commun

	Exercice : 00
colle0X	
Dossier de rendu : <i>ex00/</i>	
Fichiers à rendre : <code>main.c</code> , <code>ft_putchar.c</code> , <code>colle0X.c</code>	
Fonctions Autorisées : <code>write</code>	
Remarques : n/a	

- Les fichiers à rendre seront le `main.c`, un `ft_putchar.c` et votre `colle0X.c`, où 0X correspondra au numéro de la colle. Par exemple, `colle00.c`.
- Exemple de `main.c` :

```
int    main()
{
    colle(5, 5);
    return (0);
}
```

- Vous devrez donc écrire la fonction `colle` prenant en paramètre deux variables de type entier nommées respectivement `x` et `y`.
- Votre fonction `colle` devra afficher à l'écran un rectangle de `x` caractères de largeur, et `y` caractères de hauteur.
- Votre `main` sera modifié en soutenance pour pouvoir changer les paramètres de l'appel à la fonction `colle`. Par exemple, ce genre de choses sera testé :

```
int    main()
{
    colle(123, 42);
    return (0);
}
```

# Chapitre IV

## Colle 00

- colle(5,3) affichera ceci :

```
$> ./a.out
o---o
|   |
o---o
$>
```

- colle(5, 1) ceci :

```
$> ./a.out
o---o
$>
```

- colle(1, 1) ceci :

```
$> ./a.out
o
$>
```

- colle(1, 5) ceci :

```
$> ./a.out
o
|
|
|
|
o
$>
```

- colle(4, 4) ceci :

```
$> ./a.out  
o--o  
|  |  
|  |  
o--o  
$>
```



# Chapitre V

## Colle 01

- colle(5,3) affichera ceci :

```
$>./a.out
/***\
*   *
\***/
$>
```

- colle(5, 1) ceci :

```
$>./a.out
/***\
$>
```

- colle(1, 1) ceci :

```
$>./a.out
/
$>
```

- colle(1, 5) ceci :

```
$>./a.out
/
*
*
*
\
$>
```

- colle(4, 4) ceci :

```
$>./a.out
/**\
*  *
*  *
\**/
$>
```

# Chapitre VI

## Colle 02

- colle(5,3) affichera ceci :

```
$> ./a.out
ABBBB
B  B
CBBBC
$>
```

- colle(5, 1) ceci :

```
$> ./a.out
ABBBB
$>
```

- colle(1, 1) ceci :

```
$> ./a.out
A
$>
```

- colle(1, 5) ceci :

```
$> ./a.out
A
B
B
B
B
C
$>
```

- colle(4, 4) ceci :

```
$> ./a.out
ABBA
B  B
B  B
CBBC
$>
```

# Chapitre VII

## Colle 03

- colle(5,3) affichera ceci :

```
$> ./a.out  
ABBBB  
B   B  
ABBBB  
$>
```

- colle(5, 1) ceci :

```
$> ./a.out  
ABBBB  
$>
```

- colle(1, 1) ceci :

```
$> ./a.out  
A  
$>
```

- colle(1, 5) ceci :

```
$> ./a.out  
A  
B  
B  
B  
A  
$>
```

- colle(4, 4) ceci :

```
$> ./a.out  
ABBB  
B   B  
B   B  
ABBB  
$>
```

# Chapitre VIII

## Colle 04

- colle(5,3) affichera ceci :

```
$> ./a.out  
ABBBB  
B  B  
CBBBA  
$>
```

- colle(5, 1) ceci :

```
$> ./a.out  
ABBBB  
$>
```

- colle(1, 1) ceci :

```
$> ./a.out  
A  
$>
```

- colle(1, 5) ceci :

```
$> ./a.out  
A  
B  
B  
B  
C  
$>
```

- colle(4, 4) ceci :

```
$> ./a.out  
ABBB  
B  B  
B  B  
CBB  
$>
```