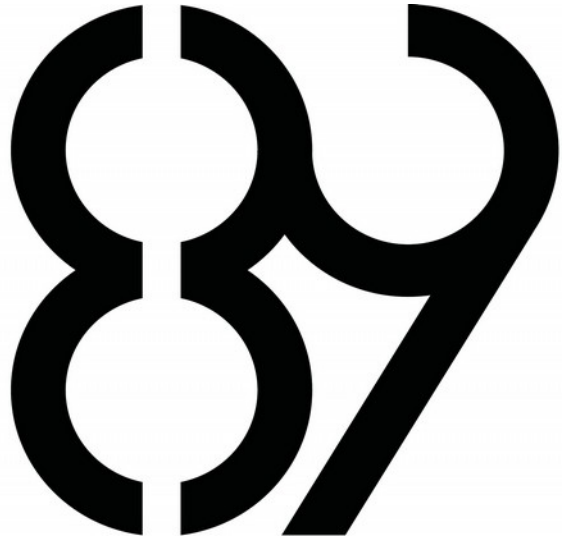




D A E M O N L A B



Palindrome

Engage le jeu que je le gagne

- DaemonLab -
infosphere@ecole-89.com

Cette colle consiste à dire si le mot passé en paramètre est un palindrome ou non.

Ce document est strictement personnel et ne doit en aucun cas être diffusé.



INDEX

Avant-propos :

- 01 – Détails administratifs
- 02 – Propreté de votre rendu
- 03 – Règlement quant à la rédaction du code C
- 04 – Construction partielle de votre rendu
- 05 – Fonctions interdites

Travail:

- 06 – Mot
- 07 – Phrase
- 08 – Anagramme



01 – Détails administratifs

Votre travail doit être envoyé via l'interface de ramassage de l'**Infosphere**.

Pour cette activité, vous rendrez votre travail sous la forme d'une archive au format tar.gz. Cette archive devra contenir l'ensemble de votre travail tel que demandé dans la section 4.

Pour créer cette archive .tar.gz, il vous suffit d'utiliser la commande suivante :

```
$> tar cvfz mon_fichier.tar.gz fichier1 fichier2 fichier3
```

Le nom « mon_fichier.tar.gz » étant à remplacer par le nom que vous souhaitez donner votre fichier archive, et « fichier1 », « fichier2 », « fichier3 » par les fichiers ou dossiers que vous souhaitez mettre dans cette archive. Vous pouvez vérifier le contenu de votre archive à l'aide de la commande « **tar -t mon_archive.tar.gz** ».

Ce travail est à effectuer seul. Vous pouvez bien sûr échanger avec vos camarades, néanmoins vous devez être l'auteur de votre travail. Utiliser le code d'un autre, c'est **tricher**. Et tricher annule **toutes** les médailles que vous avez reçu sur l'activité. La vérification de la triche est réalisée de la même manière que la correction : de manière **automatique**. Prenez garde si vous pensez pouvoir passer au travers.

Médailles accessibles :



Réussir à rendre :

Vous avez réussi à envoyer votre travail au système de correction.



02 – Propreté de votre rendu

Votre rendu, c'est à dire le contenu de l'archive ou du dépôt que vous entrez sur l'interface du TechnoCentre, doit respecter **strictement** l'ensemble des règles suivantes :

- Il ne doit contenir **aucun** fichier objet. (*.o)
- Il ne doit contenir **aucun** fichier tampon. (*.~, #*#)
- Il ne doit pas contenir votre production finale (programme ou bibliothèque)

La présence d'un fichier interdit mettra
immédiatement fin à votre évaluation.

Médailles accessibles :



Rendu propre

Votre rendu respecte les règles de
propretés imposées.



03 – Règlement quant à la rédaction du code C

En général, le code source de votre programme doit impérativement respecter un ensemble de règles de mise en page définie par les **Table de la Norme**.

Pour cette première activité, nous vous libérons de cette contrainte qui vous sera néanmoins très bientôt imposée pour l'ensemble de vos programmes écrits en C.



04 – Construction partielle de votre rendu

Le programme de correction va construire une sous-partie déterminée de votre rendu afin d'effectuer des tests dessus. En voici les paramètres :

- Les fichiers qui seront compilés sont ceux qui auront l'extension *.c.
- Seuls les fichiers dans le(s) dossier(s) ./ seront compilés.
- Les fichiers seront compilés sans règle de compilation particulière.
- Les fichiers seront compilés séparément.

L'ensemble du code que vous rendez doit pouvoir être compilé.
En cas d'échec de la compilation, vous ne serez pas évalué.

Médailles accessibles :



Construction partielle

Les éléments requis de votre projet se construisent séparément.



05 – Fonctions interdites

Vous n'avez le droit à aucune fonction.

L'utilisation d'une fonction interdite est assimilée à de la triche.
La triche provoque l'arrêt de l'évaluation et la perte des médailles.



06 – Mot

Écrivez la fonction suivante :

```
bool e89_word_palindrome(const char *str);
```

Cette fonction renvoi vrai si la chaîne de caractère passé en paramètre contient un palindrome, c'est à dire un mot qui peut se lire indifféremment dans les deux sens.

Attention : vous devez ignorer la classe, c'est à dire considérer qu'une lettre en majuscule est équivalente à sa lettre en minuscule.

Exemple : **Kayak** est un palindrome.



07 – Phrase

Écrivez la fonction suivante :

```
bool e89_sentence_palindrome(const char *str);
```

Cette fonction renvoi vrai si la chaîne de caractère passé en paramètre contient un palindrome constitué potentiellement sur plusieurs mots. C'est à dire que des espaces ou de la ponctuation peuvent être situé dans la phrase et que vous devez les ignorer, de même que la casse.

Exemple : « Engage le jeu que je le gagne » est un palindrome.



08 – Anagramme

Écrivez la fonction suivante :

```
bool e89_anagram(const char *a,  
                 const char *b) ;
```

Cette fonction renvoi vrai si les deux chaînes forment un anagramme. Les caractères n'étant pas des lettres sont à ignorer.

Exemple : « Chauve souris » et « Souche a virus »