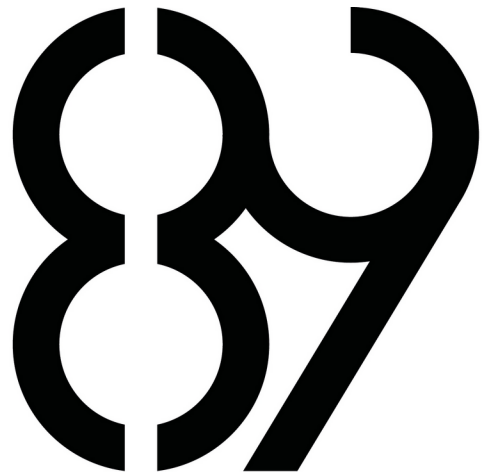




D A E M O N L A B



C-STRING

Analyse grammaticale

- DaemonLab -
daemonlab@ecole-89.com

Ce TP aborde le formatage des chaînes de caractères en C et leur parsing.

Nom de code : lcstring
Clôture du ramassage : 04/02/2020 23:59

Médailles accessibles :

Définition des médailles à venir

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 – C-String



01 – Détails administratifs

Votre travail doit être envoyé via l'interface de ramassage du **TechnoCentre** :

<http://technocentre.ecole89.com/ramassage>

Le numéro de code présent sur ce sujet vous est propre : vous devrez le renseigner en rendant votre travail. En cas d'erreur, votre travail ne sera pas associée à l'activité et votre travail ne sera pas ramassé.

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 – C-String

Écrivez la fonction suivante :

```
char *e89_read_cstring(const char *str)
```

Cette fonction lit dans la chaîne de caractère **str** les caractères qui s'y trouvent et en interprète les symboles afin d'en retourner une copie où ceux là ont été interprétés.

Par exemple, la chaîne suivante :

« Coucou\n »

Soit, en hexadecimal 0x43 0x6F 0x75 0x63 0x6F 0x75 0x5C 0x6E 0x00

Doit être transformé en :

0x43 0x6F 0x75 0x63 0x6F 0x75 0x0A 0x00

Soit une transformation de symboles '****' **antislash** et '**n**' **lettre n** en '**\n**' **nouvelle ligne**.

La liste des symboles à transformer est la suivante :

'\a', '\b', '\t', '\n', '\v', '\f', '\r'

Ainsi que :

'\0xxx' ou xxx est un nombre en **octal**.

Et :

'\xXX' ou XX est un nombre en **hexadecimal**.

Écrivez la fonction suivante :

```
char *e89_write_cstring(const char *str)
```

Cette fonction effectue l'opération inverse. Elle transforme 0x0A en 0x5C 0x6E par exemple.