



PROJET QUIZ CHALLENGE - CHAÎNES DE CARACTÈRES



OBJECTIF

Créer un QUIZ /QCM à choix multiples sur les chaînes de caractères en C.
L'utilisateur choisit une réponse (A, B, C, D), le programme utilise `strcmp()` pour comparer les chaînes, et propose un système de hints avec réessai.

NIVEAU : DÉBUTANT CHALLENGE

- Connaît les bases : `printf()`, tableaux...
 - Découvre les chaînes de caractères
 - Prêt pour un défi plus technique avec `strcmp()` !
-

FONCTIONNALITÉS CHALLENGE

Ce que fait le programme :

1. **Affiche une question** avec 4 choix (A, B, C, D)
2. **Lit la réponse** avec `gets()` (*chaîne de caractères*)
3. **Compare avec** `strcmp()` les chaînes "A", "B", "C", "D"
4. **Si c'est bon** : "Excellent !" + point + question suivante
5. **Si c'est faux** :
 - 1er essai : "Incorrect ! Hint : [indice]"
 - 2ème essai : "Toujours faux ! Réponse : [réponse]"
 - Passe à la suivante...
6. **Gestion des majuscules/minuscules** avec `strcmp()`
7. **À la fin** : score avec commentaires personnalisés



Exemple d'utilisation :

```
=== QUIZ CHALLENGE - CHAÎNES DE CARACTÈRES C ===
```

```
Question 1/7: Quelle fonction permet de lire une chaîne avec espaces ?
```

- A) scanf()
- B) gets()
- C) getchar()
- D) printf()

```
Votre choix (tapez A, B, C ou D): B
```

```
✓ Excellent ! +1 point
```

```
Question 2/7: Quelle fonction compare deux chaînes de caractères ?
```

- A) compare()
- B) strcmp()
- C) strcasecmp()
- D) equal()

```
Votre choix (tapez A, B, C ou D): A
```

```
× Incorrect ! Hint: Cette fonction commence par "str" et finit par "cmp"
```

```
Votre choix (tapez A, B, C ou D): C
```

```
✓ Correct au 2ème essai ! +0.5 point
```

```
Question 3/7: Comment déclarer un tableau de 50 caractères ?
```

- A) char nom[50]
- B) string nom[50]
- C) char nom(50)
- D) text nom[50]

```
Votre choix (tapez A, B, C ou D): a
```

```
✓ Parfait ! +1 point (accepte minuscules)
```

```
[...autres questions...]
```

```
=== RÉSULTAT FINAL ===
```

```
Score: 5.5/7 (79%)
```

```
👏 Très bien ! Tu maîtrises les bases des chaînes !
```

```
Points forts: Déclarations, fonctions de base
```

```
À améliorer: Fonctions avancées, gestion mémoire
```



PLANNING CHALLENGE

2 JOURS - 6 HEURES

Jour 1 : Structure avec `strcmp()` (3h)

- ☐ Créer quiz_chaines.c
- ☐ Système de questions avec tableaux 2D
- ☐ Utilisation de `gets()` pour lire les réponses
- ☐ Comparaison avec `strcmp()` pour "A", "B", "C", "D"

Jour 2 : Fonctionnalités avancées (3h)

- ☐ Système de hints (indices)
- ☐ Gestion majuscules/minuscules
- ☐ Score avec demi-points
- ☐ Commentaires personnalisés

SPÉCIFICATIONS TECHNIQUES CHALLENGE

Contraintes CHALLENGE

- **Maximum 200 lignes** de code
- **7 questions** minimum
- **Utilisation obligatoire** : `gets()`, `puts()`, `strcmp()`, `strcpy()`, `strlen()`
- **Gestion des majuscules/minuscules** avec `strcmp()` et conversion
- **Tableaux 2D** pour stocker questions/choix/hints

LIVRABLES CHALLENGE

Un seul fichier :

- ☐ **quiz_chaines.c** - Code source complet utilisant `strcmp()`

Ce projet challenge QCM te fera maîtriser les chaînes de caractères en C.