

## Introduction à l'architecture des ordinateurs

### TP 1 : Codage de caractères

**Note:** L'objectif de ce TP est de vous permettre d'explorer les différences entre les codages de caractères (ISO-8859-15, UTF-8, UTF-16, ASCII). Il est à faire idéalement sous Linux.

Vous prendrez soin à la fin du TP de rendre les réponses aux questions sous forme de PDF. Le rendu doit se faire via l'espace de rendu de TP sur moodle, qui correspond à votre groupe.

#### Partie 1 : Codage de caractères simples

1. Créez deux fichiers texte nommés `"iso.txt"`, `"utf8.txt"` à l'aide de votre éditeur de texte préféré (comme **Gedit**).
2. Dans chaque fichier, copiez la phrase **"Welcome to BUT1"**. Assurez-vous que chaque fichier utilise un codage différent : **ISO-8859-15** (latin-9) pour `"iso.txt"`, **UTF-8** et `"utf8.txt"`.
3. Utilisez la commande `xxd -g 1` pour afficher les représentations hexadécimales des fichiers :

```
xxd -g 1 iso.txt
```

```
xxd -g 1 utf8.txt
```

4. Comparez les représentations hexadécimales des fichiers. Est-ce que vous constatez une différence entre les deux codages ? Y a-t-il des caractères dont le codage diffère de l'ASCII ?
5. Utilisez `du` pour afficher la taille en octets (bytes) de chaque fichier. Y a-t-il une différence de taille entre les deux fichiers ? Justifiez.

#### Partie 2 : Codage de caractères spéciaux

1. Créez deux nouveaux fichiers texte nommés `"iso_special.txt"` et `"utf8_special.txt"`.
2. Dans chaque fichier, copiez la phrase : **"Je t'ai donné 1 €!"**.
3. Utilisez la commande `xxd -g 1` pour afficher les représentations hexadécimales des fichiers :

```
xxd -g 1 iso_special.txt
```

```
xxd -g 1 utf8_special.txt
```

4. Comparez les représentations hexadécimales des fichiers. Est-ce qu'il y a des caractères ayant des codages différents ? Si oui, quels sont ces caractères ? Indiquez leurs codes

hexadécimaux pour chaque codage. Y a-t-il des caractères dont le codage diffère de l'**ASCII** ? Pourquoi certains octets ne sont pas affichable en **ASCII** ?

5. Utilisez **du** pour afficher la taille en octets (bytes) de chaque fichier. Y a-t-il une différence de taille entre les deux fichiers ? Si oui expliquez la différence ?
6. Ouvrez le fichier "**utf8\_special.txt**" avec l'éditeur de texte **Gedit** en spécifiant l'encodage **ISO-8859-15** en utilisant la commande suivante :

```
gedit --encoding=ISO-8859-15 utf8_special.txt
```

Ouvrez également "**iso\_special.txt**" avec la commande :

```
gedit --encoding=UTF-8 iso_special.txt
```

Dans **Gedit**, remarquez comment le texte est interprété différemment en utilisant les codages **UTF-8** et **ISO-8859-15** (latin-9) et relevez les conséquences d'une interprétation erronée.

### **Partie 3 : Codage de caractères en UTF-16**

1. Créez un nouveau fichier texte nommé "**utf16.txt**".
2. Copiez la phrase "**Je t'ai donné 1 €!**" dans le fichier "**utf16.txt**".
3. Utilisez la commande **xxd -g 2** pour afficher les représentations hexadécimales des caractères du fichier **UTF-16** :

```
xxd -g 2 utf16.txt
```

4. Observez les représentations hexadécimales des caractères dans le fichier **UTF-16**. Comparez les représentations hexadécimales des caractères du fichier **UTF-16** avec celles du fichier **UTF-8**. Qu'est ce que vous constatez ?
5. Quels sont les caractères codés différemment entre **UTF-16** et **UTF-8** ?
6. Indiquez si le codage est en **big-endian** ou en **little-endian**. Donnez les codes des caractères **é** et **€** en décimal.
7. (**facultatif**) Vous verrez probablement les octets "**fffe**" au début du fichier **UTF-16** lorsque vous utilisez la commande **xxd**. Expliquez à quoi sert le "**fffe**" au début du fichier.

---

## Annexe : Éditeur de Texte et Commandes Utiles

### Éditeur de Texte :

**Gedit** : **Gedit** est un éditeur de texte graphique sous Linux. Il offre une interface utilisateur conviviale pour créer, éditer et afficher des fichiers texte. Pour ouvrir un fichier avec **Gedit** en spécifiant l'encodage (par exemple, **ISO-8859-15**) :

```
gedit --encoding=ISO-8859-15 nom_du_fichier.txt
```

### Commandes Utiles :

- **xxd -g 1** : Cette commande est utilisée pour afficher une représentation hexadécimale d'un fichier texte avec une sortie groupée par 1 octet (caractère). Exemple d'utilisation :

```
xxd -g 1 fichier.txt
```

Cette commande est utile pour visualiser les représentations hexadécimales des fichiers texte.

- **du** : Utilisée pour afficher l'utilisation de l'espace disque par des fichiers et répertoires. Exemple d'utilisation pour afficher la taille d'un fichier en octets : **du -b fichier.txt**