Encodage des caractères Unicode en UTF-8

Site Internet : **www.gecif.net**	Type de document : **Cours**	Intercalaire :	Date :

I – Jeu de caractères Unicode et algorithme UTF-8

Le jeu de caractères **Unicode** est universel et définitif : il contient tous les alphabets connus à ce jours et pour toujours. Il contient en plus des milliers de nouveaux symboles ou pictogrammes, avec plus de 100 000 caractères en tout aujourd'hui. Chaque caractère possède un code numérique unique appelé **point de code**. Le point de code est un nombre entre 7 et 21 bits (2 millions de possibilités) et il se note **U+xxxx** avec xxxx un nombre en hexadécimal.

UTF-8 est l'algorithme permettant d'encoder un caractère de la table Unicode. Le code numérique est alors écrit sur 1 à 4 octets. UTF-8 permet de convertir un point de code (valeur entre 0 et 2 000 000 en décimal) en une suite d'octets contenant 1 à 4 octets. Les caractères les plus courants se codent sur 1 octet, les caractères plus rares se codent entre 2 et 4 octets. Le décodage UTF-8 consiste à revenir au point de code à partir de la suite d'octets.

➔ si le point de code est **inférieur à 128** il est converti en 1 octet ayant la même valeur (de 0 à 127 : MSB à 0). Le codage de ces 128 premiers caractères est identique entre la table ASCII d'origine et l'UTF-8.

➔ si le point de code est **supérieur ou égal à 128** il est converti en une séquence d'octets de 2, 3 ou 4 octets ayant chacun une valeur entre 128 et 255 (MSB à 1) selon le principe résumé dans le tableau suivant :

Intervalle du point de code		**Taille du point de code**	**Format de l'UTF-8**	**Encodage UTF-8 (en gras les préfixes des octets)**
en décimal	**en hexadécimal**			
de 0 à 127	de 00 à 7F	7 bits	1 octet	**0**xxxxxxx
de 128 à 2047	de 80 à 7FF	8 à **11 bits**	2 octets	**110**xxxxx **10**xxxxxxx
de 2048 à 65535	de 800 à FFFF	12 à **16 bits**	3 octets	**1110**xxxx **10**xxxxxxx **10**xxxxxxx
65536 et plus	10000 et plus	17 à **21 bits**	4 octets	**11110**xxx **10**xxxxxxx **10**xxxxxxx **10**xxxxxxx

Encodage UTF-8 : **conversion UNICODE ➔ UTF-8**

Exemple sur 2 octets : En UTF-8 comment se code le caractère unicode de point de code U+05E4 ?

1. On convertit $05E4_{(16)}$ en binaire naturel **sur 11 bits** : $10111100100_{(2)}$
2. On préfixe les 5 bits de poids fort par **110** et les 6 bits de poids faible par **10** : **110**10111 **10**100100
3. Puis on convertir en hexadécimal les 2 octets obtenus : 0xD7 0xA4

Exemple sur 3 octets : En UTF-8 comment se code le caractère unicode de point de code U+79C1 ?

1. On convertit $79C1_{(16)}$ en binaire naturel **sur 16 bits** : $0111100111000001_{(2)}$
2. On sépare les 16 bits en 4+6+6 bits préfixés par **1110** et **10** : **1110**0111 **10**100111 **10**000001
3. Puis on convertir en hexadécimal les 3 octets obtenus : 0xE7 0xA7 0x81

Exemple sur 4 octets : En UTF-8 comment se code le caractère unicode de point de code U+18A3E ?

1. On convertit $18A3E_{(16)}$ en binaire naturel **sur 21 bits** : $000011000101000111110_{(2)}$
2. On répartit les 21 bits dans les 4 octets préfixés : **11110**000 **10**011000 **10**101000 **10**111110
3. Puis on convertir en hexadécimal les 4 octets obtenus 0xF0 0x98 0xA8 0xBE

A retenir : l'encodage UTF-8 consiste à convertir en binaire naturel le point de code puis à **insérer les préfixes**.

Décodage UTF-8 : **conversion UTF-8 ➔ UNICODE**

Exemple sur 2 octets : Quel est le point de code du caractère unicode qui s'encode 0xC6 0xA2 en UTF-8 ?

1. On convertit les 2 octets 0xC6 0xA2 en binaire et on reconnaît les préfixes : **110**00110 **10**100010
2. On élimine les préfixes **110** et **10** et on rassemble les 11 bits du point de code : $00110100010_{(2)}$
3. On convertit le point de code en hexadécimal sur 4 chiffres : U+01A2

Exemple sur 3 octets : Quel est le point de code du caractère unicode qui s'encode 0xEC 0x9E 0x8A en UTF-8 ?

1. On convertit les 3 octets en binaire et on reconnaît les préfixes : **1110**1100 **10**011110 **10**001010
2. On élimine les préfixes **1110** et **10** et on rassemble les 16 bits du point de code : 1100011110001010$_{(2)}$
3. On convertit le point de code en hexadécimal sur 4 chiffres : U+ C78A

Exemple sur 4 octets : Quel est le point de code du caractère qui s'encode 0xF0 0xA6 0xBD 0x8C en UTF-8 ?

1. On convertit les 4 octets en binaire : **11110**000 **10**100110 **10**111101 **10**001100
2. On élimine les préfixes et on rassemble les 21 bits du point de code : 000100110111101001100$_{(2)}$
3. On convertit le point de code en hexadécimal : U+26F4C

A retenir : le décodage UTF-8 consiste à convertir en binaire les octets de l'UTF-8 puis à **supprimer les préfixes**.

II – Exercices d'application

Encodage UTF-8 : **conversion UNICODE ➔ UTF-8**

EXERCICE 1 : En UTF-8 comment se code le caractère unicode de point de code U+05B8 ?

1. je convertis 05B8$_{(16)}$ en binaire naturel **sur 11 bits** : ……$_{(2)}$
2. je préfixe les 5 bits de poids fort par **110** et les 6 bits de poids faible par **10** : **110**…… **10**……
3. je convertis en hexadécimal les 2 octets obtenus : 0x…… 0x……

EXERCICE 2 : En UTF-8 comment se code le caractère unicode de point de code U+05E4 ?

1. je convertis 05E4$_{(16)}$ en binaire naturel **sur 11 bits** : ……$_{(2)}$
2. je préfixe les 5 bits de poids fort par **110** et les 6 bits de poids faible par **10** : …… ……
3. je convertis en hexadécimal les 2 octets obtenus : 0x…… 0x……

EXERCICE 3 : En UTF-8 comment se code le caractère unicode de point de code U+8B0A ?

1. je convertis 8B0A$_{(16)}$ en binaire naturel **sur 16 bits** : ……$_{(2)}$
2. je répartis les 16 bits sur 3 octets en insérant les préfixes : **1110**…… **10**…… **10**……
3. je convertis en hexadécimal les 3 octets obtenus : 0x…… 0x…… 0x……

EXERCICE 4 : En UTF-8 comment se code le caractère unicode de point de code U+2767 ?

1. je convertis 2767$_{(16)}$ en binaire naturel **sur 16 bits** : ……$_{(2)}$
2. je répartis les 16 bits sur 3 octets en insérant les préfixes : …… …… ……
3. je convertis en hexadécimal les 3 octets obtenus : 0x…… 0x…… 0x……

Décodage UTF-8 : **conversion UTF-8 ➔ UNICODE**

EXERCICE 5 : Quel est le point de code du caractère unicode qui s'encode 0xD6 0xB9 en UTF-8 ?

1. je convertis les 2 octets 0xD6 0xB9 en binaire et je reconnais les préfixes : **110**…… **10**……
2. j'élimine les préfixes **110** et **10** et je rassemble les 11 bits du point de code : ……$_{(2)}$
3. je convertis le point de code en hexadécimal sur 4 chiffres : U+……

EXERCICE 6 : Quel est le point de code du caractère unicode qui s'encode 0xD5 0x8F en UTF-8 ?

1. je convertis les 2 octets 0xD5 0x8F en binaire et je reconnais les préfixes : …… ……
2. j'élimine les préfixes et je rassemble les 11 bits du point de code : ……$_{(2)}$
3. je convertis le point de code en hexadécimal sur 4 chiffres : U+……

EXERCICE 7 : Quel est le point de code du caractère unicode qui s'encode 0xE2 0x95 0xA5 en UTF-8 ?

1. je convertis les 3 octets en binaire et je reconnais les préfixes : …… …… ……
2. j'élimine les préfixes et je rassemble les 16 bits du point de code : ……$_{(2)}$
3. je convertis le point de code en hexadécimal sur 4 chiffres : U+……