



ISN – Informatique et Sciences du Numérique

TP CODAGE DE L'INFORMATION



1 – CODAGE D'UN NOMBRE

1.2 – LE SYSTEME BINAIRE

1.2.1 – Conversion binaire vers décimal

Exercice n°1

$$(1110)_2 = 14 \quad (1011 \ 1001)_2 = 185$$

1.2.2 – Conversion décimal vers binaire

Exercice n°2

$$219 = (1101 \ 1011)_2 \quad 186 = (1011 \ 1010)_2$$



1.3 – LE SYSTEME HEXADECIMAL

1.3.1 – Conversion hexadécimal vers décimal

Exercice n°3

$$(5D)_{16} = 93 \quad (F3C)_{16} = 3900$$

1.3.2 – Conversion décimal vers hexadécimal

Exercice n°4

$$184 = (B8)_{16} \quad 252 = (FC)_{16}$$

1.3.3 – Conversion binaire vers hexadécimal

Exercice n°5

$$(F3)_{16} = (1111\ 0011)_2 \quad (45A)_{16} = (0100\ 0101\ 1010)_2$$

1.3.4 – Conversion hexadécimal vers binaire

Exercice n°6

$$(1001\ 1011)_2 = (9B)_{16} \quad (1000\ 1001)_2 = (89)_{16}$$

1.5 – NOMBRES RELATIFS (ENTIERS SIGNES)

Exercice n°7

1. $36 = (0010\ 0100)_2$

$$\begin{array}{rcl}
 115 & \rightarrow & 1\ 1\ 1\ 0\ 0\ 1\ 1 \\
 \text{Complément} & \rightarrow & 0\ 0\ 0\ 1\ 1\ 0\ 0 \\
 & + & \underline{\hspace{1.5cm}1} \\
 \text{Complément à 2} & \rightarrow & 0\ 0\ 0\ 1\ 1\ 0\ 1
 \end{array}$$

$$-115 = (1000\ 1101)_2$$



Exercice n°9 : Programmes de conversion

```
def equDH(n10):
# Equivalence Decimal Hexadecimal
    if (n10 == 10) : n16 = "A"
    elif(n10 == 11) : n16 = "B"
    elif(n10 == 12) : n16 = "C"
    elif(n10 == 13) : n16 = "D"
    elif(n10 == 14) : n16 = "E"
    else : n16 = "F"
    return n16

def equHD(n16):
# Equivalence Hexadecimal Decimal
    if (n16 == "A") : n10 = 10
    elif(n16 == "B") : n10 = 11
    elif(n16 == "C") : n10 = 12
    elif(n16 == "D") : n10 = 13
    elif(n16 == "E") : n10 = 14
    elif(n16 == "F") : n10 = 15
    else : n10 = int(n16)
    return n10

def convBH(n2) :
# Conversion Binaire Hexadecimal
    n = len(n2)
    n10 = i = 0
    n16=""
    while(i<n):
        for j in range (0,4) :
            n10 = n10+int(n2[n-1-i-j])*2**j
            if (n10>9): quartet = equDH(n10)
            else : quartet = str(n10)
            n16 = quartet + n16
            i = i+4
            n10 = 0
        return n16

def convBD(n2) :
# Conversion Binaire Decimal
    n10 = 0
    n = len(n2)
    for i in range (0,n) :
        n10 = n10+int(n2[i])*2**(n-1-i)
    return n10

def convDB(n10) :
# Conversion Decimal Binaire
    n2=""
    while (n10 >=2) :
        bit = n10 % 2;
        print("bit",bit)
        n2 = str(bit) + n2
        print(n2)
        n10 = n10 // 2;
    n2 = "1" + n2
    print(n2)
    return n2

def convDH(n10) :
# Conversion Decimal Hexadecimal
    n16=""
    while (n10 >=16) :
        symbole = n10 % 16;
        if (symbole>9): symbole = equDH(symbole)
        else : symbole = str(symbole)
        n16 = symbole + n16
        n10 = n10 // 16;
    if (n10>9): n10 = equDH(n10)
    else : n10 = str(n10)
    n16 = n10 + n16
    return n16

def convHB(n16) :
# Conversion Hexadecimal Binaire
    n2=""
    n = len(n16)
    for i in range(0,n) :
        n10 = equHD(n16[i])
        quartet = convDB(n10)
        if(len(quartet)==0) : quartet = "0000"
        elif(len(quartet)== 1) : quartet = "000" + quartet
        elif(len(quartet)== 2) : quartet = "00" + quartet
        elif(len(quartet)== 3) : quartet = "0" + quartet
        n2 = n2 + quartet
    return n2
```



```
def convHD(n16) :
# Conversion Hexadecimal Decimal
    n10 = 0
    n = len(n16)
    for i in range (0,n) :
        n10 = n10+equHD(n16[i])*16**(n-1-i)
    return n10

-----
# Programme principal
-----
dep = input("Base de depart Binaire (2), Decimale (10) ou Hexdecimale (16)")

# Le nombre de depart est binaire
if (dep == "2"):
    arr = input("Vers quelle base : Decimale (10) ou Hexdecimale (16) ?")

    # Conversion binaire decimal
    if (arr == "10") :
        print("Conversion Binaire --> Decimal")
        N2=input("Entrer le nombre binaire")
        N10 = convBD(N2)
        print("L'equivalent decimal de",N2,"est",N10)

    # Conversion binaire hexadecimal
    else :
        print(" Conversion Binaire -- >Hexadecimal")
        N2=input("Entrer le nombre binaire")
        N16 = convBH(N2)
        print("L'equivalent Hexadecimal de",N2,"est",N16)

# Le nombre de depart est decimal
elif (dep == "10"):
    arr = input(" Vers quelle base : Binaire (2) ou decimale (16) ? ")

    # Conversion decimal binaire
    if (arr == "2") :
        print(" Conversion Decimal -- > Binaire ")
        N10=int(input("Entrer un nombre decimal"))
        N2 = convDB(N10)
        print("L'equivalent binaire de",N10,"est",N2)

    # Conversion decimal hexadecimal
    else :
        print("Conversion Decimal --> Hexadecimal")
        N10=int(input("Entrer un nombre decimal"))
        N16 = convDH(N10)
        print("L'equivalent hexadecimal de",N10,"est",N16)

# Le nombre de depart est hexadecimal
else :
    arr = input(" Vers quelle base : Binaire (2) ou Decimale (10) ? ")

    # Conversion hexadecimal binaire
    if (arr == "2") :
        print("Conversion Hexadecimal --> Binaire")
        N16=input("Entrer un nombre hexadecimal")
        N2 = convHB(N16)
        print("L'equivalent binaire de",N16,"est",N2)

    # Conversion hexadecimal decimal
    else :
        print("Conversion Hexadecimal --> Decimal")
        N16=input("Entrer un nombre hexadecimal")
        N10 = convHD(N16)
        print("L'equivalent binaire de",N16,"est",N10)
```



2 – CODAGE BINAIRE D'UN TEXTE

2.1 – PROBLEMATIQUE

Exercice n°10

1. Le **codage** de certains caractères **n'est pas correct**. Le mode de codage des caractères utilisé pour l'affichage du texte n'est pas identique à celui qui a été utilisé pour son édition.
2. Dans le texte présenté, le problème concerne les **lettres accentuées**.

2.2 – LE CODE ASCII

Exercice n°11

1.

L	a		c	o	m	p	a	g	n	i	e		d	e
0x4C	0x61	0x20	0x63	0x6F	0x6D	0x70	0x61	0x67	0x6E	0x69	0x65	0x20	0x64	0x65
l	'	a	n	n	e	a	u							
0x6C	0x60	0x61	0x6E	0x6E	0x65	0x61	0x75							

2.

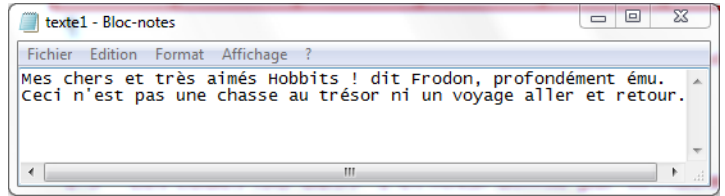
0x46	0x72	0x6F	0x6E	0x64	0x6F	0x6E	0x20	
01000110	01110010	01101111	01101110	01100100	01101111	01101110	00100000	
F	r	o	n	d	o	n		
0x6C	0x65	0x20	0x48	0x6F	0x62	0x62	0x69	0x74
01101100	01100101	00100000	01001000	01101111	01100010	01100010	01101001	01110100
l	e		H	o	b	b	i	t

3. Le code ASCII **ne permet pas de coder les lettres accentuées**.

2.3 – LES CODES ISO 8859–1 ET ANSI UTILISE PAR WINDOWS

Exercice n°12

1. La taille du fichier **texte1.txt** est de **130 octets**. Ce texte est constitué de **106 caractères**, **22 espaces(SP)** et un **saut à ligne (2 octets : LF et CR)**. Ce qui fait **130 octets** pour coder l'ensemble du texte.



3. Les octets numéro **0x3f** et **0x40** ont pour valeur **0x0D** et **0x0A** qui correspondent à l'action **passer à la ligne et retour au début (CR, LF)**.

4.

B	i	l	b	o	n
0x42	0x69	0x6C	0x62	0x6F	0x6E

5.

0000009e	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00000000	4d	65	73	20	63	68	65	72	73	20	65	74	20	74	72	e8
00000010	73	20	61	69	6d	e9	73	20	48	6f	62	62	69	74	73	20
00000020	21	20	64	69	74	20	42	69	6c	62	6f	6e	2c	20	70	72
00000030	6f	66	6f	6e	64	e9	6d	65	6e	74	20	e9	6d	75	2e	0d
00000040	0a	43	65	63	69	20	6e	27	65	73	74	20	70	61	73	20
00000050	75	6e	65	20	63	68	61	73	73	65	20	61	75	20	74	72
00000060	e9	73	6f	72	20	6e	69	20	75	6e	20	76	6f	79	61	67
00000070	65	20	61	6c	6c	65	72	20	65	74	20	72	65	74	6f	75
00000080	72	2e

```
Mes chers et très aimés Hobbits ! dit Bilbon, profondément ému.
Ceci n'est pas une chasse au trésor ni un voyage aller et retour.
```

6.

```
Mes·chers·et·très·aimés·Hobbits·l·dit·Frodon·,·profondément·ému·¶
Ceci·n'est·pas·une·chasse·au·trésor·ni·un·voyage·aller·et·retour·¶¶
```

7. La taille de ce fichier est de **24 064 octets**. Un fichier « doc » **contient d'autres informations** (styles, polices,...) que le texte brut.

9. Le **contenu est incompréhensible**. Le contenu du fichier, contient hormis le texte, d'autres informations concernant la **mise en forme** (style, modèle...).

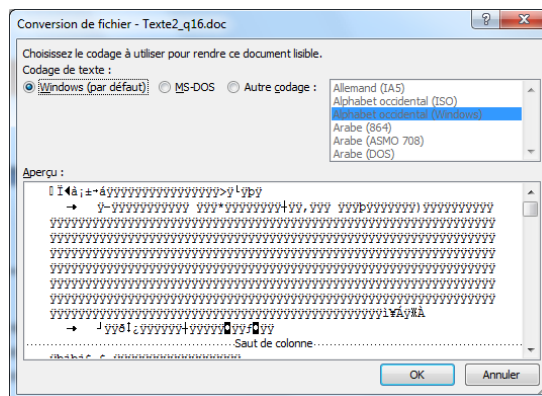
10.

```

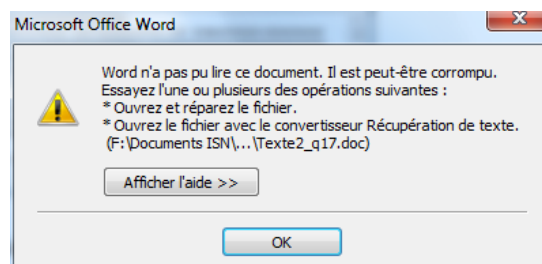
4d 65 73 20 63 68 65 72 73 20 65 74 20 74 72 e8
73 20 61 69 6d e9 73 20 48 6f 62 62 69 74 73 20
21 20 64 69 74 20 42 69 6c 62 6f 6e 2c 20 70 72
6f 66 6f 6e 64 e9 6d 65 6e 74 20 e9 6d 75 2e 0d
43 65 63 69 20 6e 27 65 73 74 20 70 61 73 20 75
6e 65 20 63 68 61 73 73 65 20 61 75 20 74 72 e9
73 6f 72 20 6e 69 20 75 6e 20 76 6f 79 61 67 65
20 61 6c 6c 65 72 20 65 74 20 72 65 74 6f 75 72
    
```

Mes-chers-et-très-aimés-Hobbits-!-dit-Bilbon,-profondément-ému.¶
 Ceci-n'est-pas-une-chasse-au-trésor-ni-un-voyage-aller-et-retour.¶

11. **Problème de codage de l'information.** L'octet numéro **0x0000** contient une information **le codage utilisé.**



12. .Il est **impossible d'ouvrir le fichier.** Un message indiquant que le **fichier est corrompu.** L'octet numéro **0x0040** est **un octet de configuration.** Une modification de cet octet entraîne une mauvaise configuration du fichier.



2.4 – CODAGE UNICODE

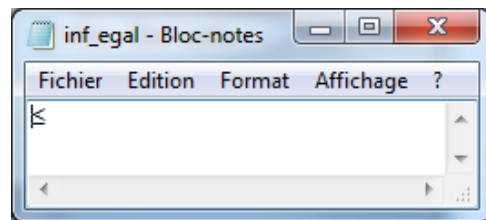
Exercice n°13

1. Dét

Valeur Unicode		Code UTF-8	
Décimale	Binaire	Binaire	Hexadécimal
8804	10 001001 100100	11100010 10001001 10100100	E2 89 A4

2.

00000013	00 01 02
00000000	e2 89 a4



5.

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
```

6.



Le **codage** de certains caractères **n'est pas correct**. Le mode de codage des caractères utilisé pour l'affichage du texte n'est pas identique à celui qui a été utilisé pour son édition.

7. Le **codage ISO 8859-1 utilise un octet par caractère**. Lorsque l'on change l'encodage les caractères « é » (e accent aigu) qui est codé sur 3 octets est remplacé par 3 caractères. Le caractère « ' » (guillemet) qui est codé sur 2 octets est remplacé par 2 caractères.

« é »	0xC3	0x40
	Ã	©