

```

1  def convertbin(vardec):      #On declare une fonction qui encode en
    ↪ binaire
2      liste=[]                #Ici on declare une liste vide
3      total=''                #Ce total est en caractères car le resultat
    ↪ ne devra pas etre additionné , il devra etre ajouté a la suite
4      while vardec>0:         #Cette boucle divise par 2 en gardant le
    ↪ reste dans une liste
5          liste=liste+[vardec%2]    #on stocke a la suite a chaque
    ↪ fois le resultat du modulo dans une nouvelle case de la
    ↪ liste
6          vardec=vardec//2          #Ici on effectue une division
    ↪ entiere par 2 pour pas avoir la virgule
7                                  #Il faut noter que comme pour
    ↪ encoder sur un papier , le
    ↪ resultat sera a l'envers
8                                  #On va donc le ranger dans le bon
    ↪ sens
9      compteur=len(liste)-1      # len(var) sert a avoir la
    ↪ longueur d'une variable , ici on la décrémente pour arriver
    ↪ a 0
10     while compteur>=0:         #Compteur est initialisé a la
    ↪ longueur de la liste et sera décrémentée jusqu'a 0 inclu
11         total=total+str(liste[compteur]) #Ici on copie chaque
    ↪ éléments de la liste dans une variable de type text(str)
    ↪ pour eviter d'additionner le contenu de la liste
12         #ex : si on a une liste contenant 01011001, et si on
    ↪ additionne sans le type str, on obtiendra 4
13         #ex (le bon) : si on a une liste contenant 01011001, et
    ↪ si on additionne avec le type str, on obtiendra 10011010
    ↪ dans le bon sens
14         compteur=compteur-1      #Ici on decremente le compteur
    ↪ jusqu'a 0 ref : a la condition de la boucle
15     return total               #On retourne le total qui sera
    ↪ stocké dans une variable
16
17 def convertdec(varbin):       #Fonction qui convertit en decimal
18     compteur=choix ==1        #initialise un compteur a 0
19     longueur=len(varbin)-1    #On stocke dans longueur la taille
    ↪ de varbin -1 (car l'indexation commence a 0)
20     total=0                   # on initialise le total a 0
21     while compteur<len(varbin): # Boucle qui effectue des
    ↪ operations de 0 a la longueur de varbin
22         total=total+(int(varbin[compteur])*(2**longueur)) # cf :
    ↪ formule pour encoder un binaire en decimal
23         compteur=compteur+1    # On incremente pour parcourir la
    ↪ chaine en entiere
24         longueur=longueur-1    # On décrémente pour avoir la
    ↪ puissance suivante
25     return total              # retourne le total
26
27     choix=1                   #on initialise le choix a 1 pour
    ↪ qu'il puisse deja entrer dans la boucle
28     while choix!=0:           #Tant que le choix est different
    ↪ de 0 , on continue a encoder des chiffres
29     choix=int(input("Que voulez vous faire :\nEncoder en binaire
    ↪ (1)\nEncoder en decimal (2)\nQuitter (0)\nChoix: "))
30     #On stocke dans choix , l'action que l'on va effectuer
31     #input sert a demander tout simplement une donnée a entrer
    ↪ au clavier

```

C:\Users\user\Desktop\Python\conver binaire.decimal.py

```
32     #'\n' sers a sauter une ligne
33     if choix ==1: # Si le choix est éale a 1 faire...
34         vardec=int(input("Entrez un chiffre a encoder en ↵
↳         binaire: "))
35         #On stocke dans choix le chiffre en decimal (on le force ↵
↳         a etre un entier par le biais de int(var))
36         resultat=convertbin(vardec) #On stocke dans une variable ↵
↳         resultat , le resultat de l'appel de la fonction ↵
↳         convertbin()
37         print("\nResultat: ",resultat,"\n" ) # afficher le résultat
38     elif choix==2: # Meme principe que précédemment
39         varbin=str(input("Entrez un chiffre a encoder en ↵
↳         decimal: "))
40         resultat2=convertdec(varbin)
41         print("\nResultat: ",resultat2,"\n")
```